3. FUNCTIONAL VIEW

The ICS Functional View describes the partitioning of ICS software elements, application level protocols used between the elements, and a discussion of how the elements might be arranged on physical hosts. Included is a Functional Framework which shows the arrangement of ICS elements and identifies the services provided by each element.

3.1 Architecture Foundations

This section discusses several topics which lay the ground work for the Functional Framework which is presented in the next section. This section discusses the conceptual design which was used during the requirements development; discusses the ICS as a generic three-tier architecture; discusses how the choice of Z39.50 as the underlying protocol for CIP affects ICS; and discusses how guide document management is performed using http as a base protocol.

3.1.1 ICS Domain Models

ICS Compatability: Explanatory

The ICS domain consists of two protocol domains, the CIP Domain and the ICS Guide Protocol (IGP) Domain as shown in Figure 3-1. The ICS URD [R2] defines ICS around a common protocol, CIP, and the elements which speak CIP. The URD discusses the CIP domain as a virtual 'CIP space' within which CIP messages, consisting of requests and responses, are exchanged between architectural elements: clients, servers and middleware. The CIP space is bounded by interfaces to translators, that is, software that converts CIP messages into external formats. (See also Section 2 for a discussion of CIP space.) As described in the ICS Guide Design and Protocol Specification [R24], the IGP provides document management using IGP messages that are exchanged between clients, servers, and middleware using an architectural approach that parallels CIP. Thus, the ICS Client is represented by the CIP Client and the HTTP Client; ICS middleware is represented by the CIP Retrieval Manager and Catalogue Translator along with the IGP Guide Manager and Guide Translator; and the ICS-related servers are represented by the respective local inventory, archive, and guide documents. The partitioning of functionality between the architectural components is discussed further in the next section

The URD [R2] also defines the ICS from a systems perspective in terms of providing a seamless interoperable environment whereby users can conveniently access all available data, whoever it belongs to, requiring both technical and administrative problems to be addressed by ICS. The ICS

also must provide management of data access; user authentication, user session management, problems of routing (sending client requests to the correct server) and collation (accumulation of different sets of results). This leads to the concept of middleware, which sits between the client(s) and server(s).

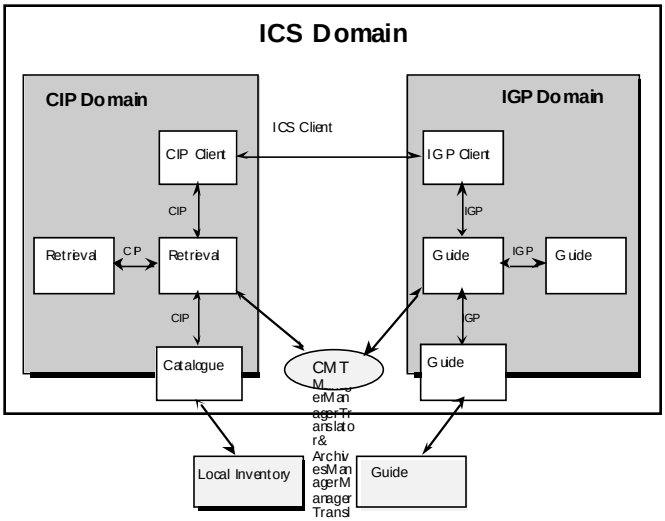

Figure 3-1. ICS Domain

The URD also comments that dividing the functionality between client, middleware, and CIP specific components of the server, is not to be resolved at the level of the URD. A range of architectural solutions can be envisaged, with different distributions of functionality and emphasis, and these should be traded off and evaluated at the level of system design analysis. These issues are addressed within this SDD.

3.1.2 Three Tier Architecture

ICS Compatability: Explanatory

The ICS approach is a prime example of a generalized concept for distributed information systems which is referred to as the Three-Tier interoperability architecture (see Figure 3-2). The three-tiered application model splits an application into its three logical component types, clients, middleware, and servers. The three tiers refer to the three logical component parts of an application, not to the number of machines used by the application. There may be any number of each of the component types within an application. Application components can be shared by any number of application systems. The application components communicate with each other using a distributed computing infrastructure. The three-tier architecture supports provider sites which manage their own resources and processes.

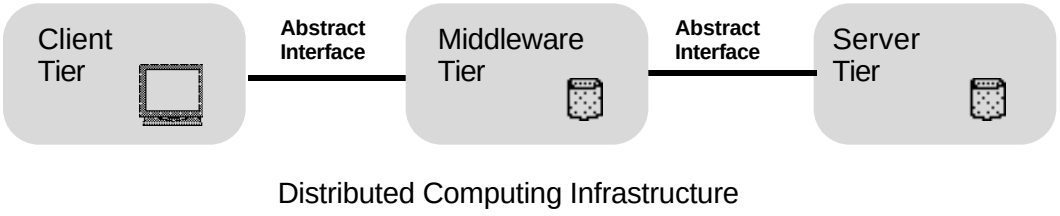

Figure 3-2. Three Tier Architecture

The Client tier consists of the user environment. A client component contains the logic which presents information to an external source and obtains input from that source. In most cases the external source is a human end user working at their own computer, although the external source might also be process-oriented. The client logic generally provides menus of options to allow the user to navigate through the different parts of the application, and it manipulates the input and output fields on the display device. Frequently, the presentation component also performs a limited amount of input data validation.

Middleware applications are entities that are used to model "business" processes. For ICS the "business" is catalogue interoperability. ICS middleware models the data and processes necessary for providing the client transparent access to heterogeneous catalogues at physically distributed locations. This is accomplished through interaction with the communication structure and an abstract interface, i.e., CIP.

The Server tier contains the logic which interfaces either with a data storage system, or with some other type of external data source such as a data feed or an external application system. Server functions to a data provider are generally invoked by the middleware, although in some applications they may be invoked directly by a presentation component.

A common mistake made in planning for three-tier architectures is assuming that, because a client/server application has three logical tiers, it must be implemented in three physical tiers. This misconception is unfortunate because it limits opportunities to capture application-specific requirements in the areas of user platform, network support, platform price/performance, development tools, and management capabilities. The three logical components can be distributed in many different ways to provide optimum configurations for application maintenance and support. The key is that once the three-tier architecture has been embraced in the design of an application, the partitioning of that application into its physical instantiation should be done in a way that optimizes performance, security, integrity, maintainability, and management.

Distributed computing infrastructures for three tier architectures is a current topic of much research and development that promises to provide transparent, dynamic access to services. For this purpose ICS relies instead on a proven and deployed infrastructure of Internet services, including DNS directory service. These are described in Section 5.

3.1.3 Z39.50 as a Base Protocol

ICS Compatability: Explanatory

Communications using the CIP are stateful, client/server sessions based on the Z39.50 Protocol [R9]. Z39.50 is an ANSI standard for information Retrieval. The Z39.50 protocol was chosen based on analysis of the URD requirements versus the services already defined in the Z39.50-1995 specification [R10]. CIP uses a subset of the Z39.50 messages to define an EO profile. CIP provides a set of EO specific attributes, as well as an extension of Z39.50 messages, through the provision of search control commands particular to the *collections* which are defined for ICS. CIP is the interface protocol for communications for collection and product searching.

3.1.4 HTTP as a Base Protocol

Communications in the IGP domain are accomplished using a set of messages that are based on the Hypertext Transfer Protocol (HTTP) protocol. The HTTP is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a generic, stateless, object-oriented protocol which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods. A feature of HTTP is the typing and negotiation of data representation, allowing systems to be built independently of the data being transferred. HTTP allows an open-ended set of methods that indicate the purpose of a request. It builds on the discipline of reference provided by the Uniform Resource Identifier (URI), as a location (URL) or name (URN) , for indicating the resource to which a method is to be applied. Messages are passed in a format similar to that used by Internet mail as defined by the Multipurpose Internet Mail Extensions (MIME). Within the scope of the IGP Domain, IGP messages based on the HTTP protocol are passed between the HTTP Client, the Guide Manager, and the Guide Translator in order to provide guide document searching and retrieval.

3.2 ICS Functional Framework

The ICS Functional Framework provides the top-level design of the application processes which make up ICS. In this section, the context of ICS is described first. Next, the ICS Elements are introduced according to their role in the three-tier architecture introduced in the previous section. Lastly the ICS Functional Framework which shows several sites is presented.

3.2.1 Context of ICS

ICS Compatability: Explanatory

The context of ICS is defined by the interfaces to systems external to ICS. Graphically this is displayed in Figure 3-3. (Note that in the convention of context diagrams, interfaces between external items are not shown, e.g., no interface is shown between Users and CEOS Agency Systems, as these are out of the scope of ICS.) There are five types of external systems with which ICS has interfaces: *Users, CEOS Agency Systems, Web Indexers, Other CIP Based Federations, GEO Based Systems, and Other Z39.50 Based Services.*

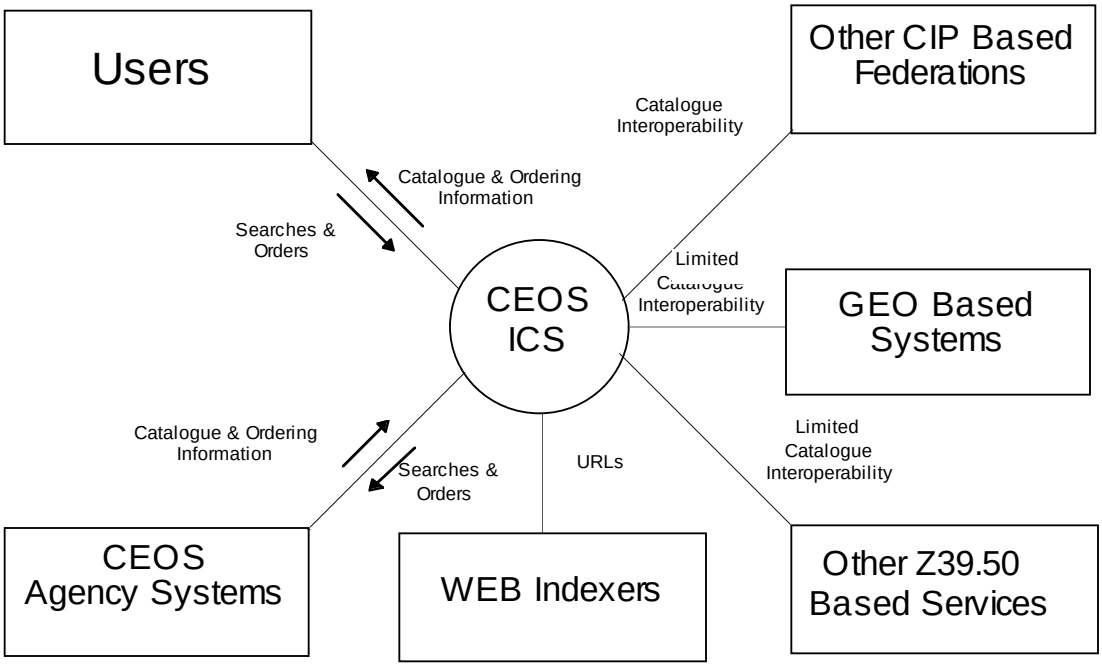

Figure 3-3. ICS Context Diagram

Users in Figure 3-3 are the human users which interface with ICS using a variety of clients. *Users* submit searches and orders to ICS and ICS responds with catalogue and ordering information respectively.

CEOS Agency Systems in Figure 3-3, are those data providers which provide catalogue services to ICS. When searches and orders are appropriately routed to the *CEOS Agency Systems*, they will respond with catalogue and ordering information respectively. *CEOS Agency Systems* includes affiliate CEOS agencies as well. Services assumed to be part of the *CEOS Agency Systems* are a catalogue service, order handling, user profiling, archiving, and site system management services.

Web Indexers in Figure 3-3, such as Alta Vista, can access ICS document collections via URLs. *Other CIP Based Federations* in Figure 3-3, are groups of other data providers which provide catalogue interoperability using CIP. These other federations, may or may not have a design similar to ICS.

GEO Based Systems in Figure 3-3, are groups of data providers which use GEO-based Z39.50 profiles. Significant effort has been undertaken to bring CIP and GEO into a level of alignment that maximizes interoperability.

Other Z39.50 Based Services in Figure 3-3, are other groups of data providers which provide interoperability using Z39.50 but not necessarily using CIP. Catalogue interoperability with these providers will be limited to those supported by the specific Z39.50 profile which the providers are using.

3.2.2 Introduction of ICS Elements

Before describing how the ICS elements are connected to make a system, the individual elements are introduced. The elements are organized following the three tier architecture introduced earlier: Clients, Middleware and Servers.

In addition to the elements described in this section, the ICS is hosted on a variety of hardware platforms and operates over a variety of networks. These issues are considered in other sections of the SDD, e.g. Sections 5 and 8.3.

3.2.2.1 ICS Client

ICS Compatability: Explanatory

The main element of the client tier is the *ICS Client* (Figure 3-4). The *ICS Client* is used by a human user to compose searches and orders which are directed to the middleware elements. The *ICS Client* consists of a presentation layer, a local management layer, an application layer, and two communication protocol clients for CIP and IGP.

The CIP client and IGP client layers accept internal messages from other parts of the *ICS Client* and create messages formatted in the CIP and IGP protocols. Also they receive CIP and IGP messages and create internal messages. The *ICS Client* may also provide access to other protocols. For example, the client may incorporate FTP as a mechanism for accessing products made available through the ordering process.

The *application layer* manages the other parts of the *ICS client* including state management most of the user perceived functionality. The *application layer* accepts inputs from the user via the *presentation layer*. The *application layer* does all local processing on the data, e.g., query formulation, and passes a message to the CIP or IGP client for transmission to the server. The *application layer* configures the *ICS Client* using data from the *Retrieval Manager* Explain database. The *application layer* is the bridge between the CIP domain and the IGP domain. For example, a URL for a guide document (that it related to a collection) received via CIP is accessed using IGP.

The *presentation layer* deals with how the information is presented to the user, including all issues related to HMI, as well as dealing with certain format specific issues, e.g., displaying browse imagery.

The *local management layer* provides functionally for local data management, e.g. saving a result set, or converting result sets into orders.

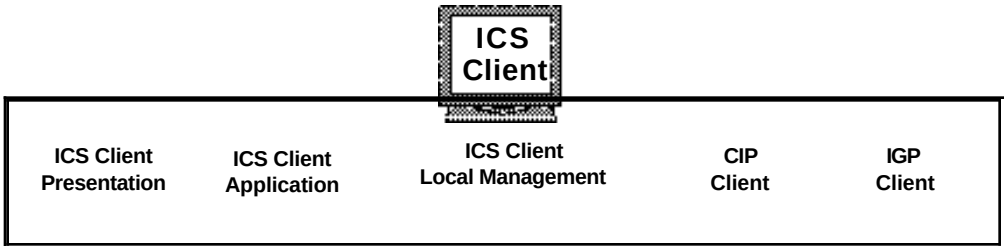

Figure 3-4. ICS Client

3.2.2.2 Retrieval Manager

ICS Compatability: Explanatory

The middleware tier provides the ICS mechanisms which enable access to the back end data providers that supply product and/or related document information. The ICS elements which comprise the middleware tier are the *Retrieval Manager* and a series of *Translators* for product and document access and the *Guide Manager* for document access.

The *Retrieval Manager* is the main element of ICS (Figure 3-5). The *Retrieval Manager* holds the *collections* which are key to the distributed searching. The *Retrieval Manager* is key as it is the common element which routes the CIP messages. Furthermore the *Retrieval Manager* enables the following key ICS services:

- dynamic configuration of clients with semantic attributes
- *collection* searching

- routing to distributed sites
- access to diverse and heterogeneous catalogues
- flexible extensions and incorporation of additional services
- generation of 'secondary' searches from original searches for theme collections
- monitoring of 'secondary' searches
- retrieval and combination of search results from 'secondary' searches

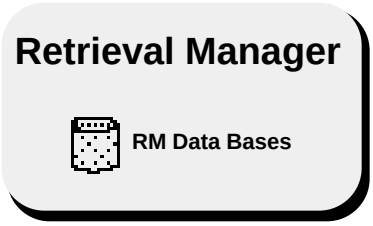

Figure 3-5. Retrieval Manager

3.2.2.3 CIP Translators

ICS Compatability: Explanatory

The second middleware element type are the *CIP translators* (See Figure 3-6). *CIP translators* provide a *Retrieval Manager* with access points to data providers. *CIP Translators* convert CIP messages into the local messages of the data provider. The portion of a *CIP translator* which handles CIP is common to other *CIP translators* across ICS. The back half of a *CIP translator* is unique to the data provider and the internal protocols that are supported by data provider.

CIP translator configurations will be defined based on the local data provider needs. Some data providers will use a single translator to convert all CIP messages to the data provider's local protocol messages. Other data providers will have multiple *CIP translators* to support different CIP messages. For the multiple translator approach the following types have been defined:

- *Catalogue Translator*
- *Order Handling System (OHS) Translator*
- *User Profile System (UPS) Translator*

Given the variety of *CIP Translator* configurations, the *CIP Translator* elements are shown generically in Figure 3-6.

If a data provider uses CIP as the internal protocol, a *Translator* is not required.

Figure 3-6. CIP Translators

3.2.2.4 Guide Manager Elements

ICS Compatability: Explanatory

The third type of middleware is the ICS Guide Manager as depicted in Figure 3-7. This consists of two components, the Guide Server and the Guide Indexer. The Guide Server interacts with the IGP Client and the Guide Translator to perform searches and retrievals for documents. It also maintains the Guide Metadata Database and the Attributes Defaults File for guide management. When a document is inserted into ICS, the Guide Indexer controls the indexing of the guide document so that free-text and fielded searches can be performed against the document via the ICS Client.

Figure 3-7. ICS Guide Manager

3.2.2.5 Guide Translator

ICS Compatability: Explanatory

Similar to the *CIP translators*, a *Guide translator* element is defined in ICS (See Figure 3-8). The Guide Translator converts document search and retrieval services for the local Guide Document Archive.

Figure 3-8. Guide Translator

3.2.2.6 Server Elements

ICS Compatability: Explanatory

The last tier of the three tier architecture includes the server elements (Figure 3-9). Server elements are the elements which manage access and storage of the EO data of interest to the users. Strictly speaking the server elements are outside of the scope of ICS because they belong to data providers and are contained in the CEOS Agency Systems block on the ICS Context Diagram. Four server elements were introduced previously: Catalogue, OHS, UPS, and Guide Document Archive. Additional elements are the Archive and the Site System Management (SSM). The Translators may not directly interact with the archive but for complete depiction of the ICS design, an archive holding the actual product data is needed. The SSM is present for those data providers which provide coordinated, on-line management of the distributed processing environment.

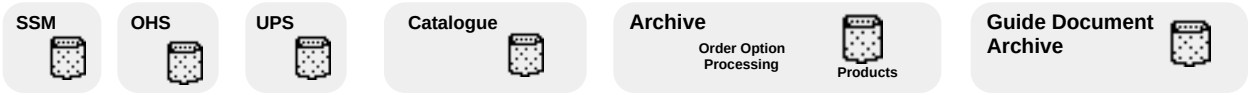

Figure 3-9. Server Elements

3.2.2.7 Other ICS Elements

ICS Compatability: Explanatory

There are other important elements which do not neatly fit into the three-tier architecture but are required for the ICS (Figure 3-10). The *ICS Site Administrator (ISA)* is the human operator that performs all tasks needed to establish and maintain a *Retrieval Manager* and, potentially, a *Guide Manager*. In practice, this is more than one person as the tasks are of various types: a scientist for *collection* definition, a database expert for maintaining the *Retrieval Manager* or *Guide Manager* databases, a system operator for diagnosis and correcting operational activities, etc. For convenience purposes all of these tasks are performed by the *ISA*.

The *Collection Management Tool (CMT)* (Figure 3-10), perform tasks involved with populating and maintaining the data in the Retrieval Manager. These tasks involve translating *collection* or directory information into CIP *collection* format and checking for valid entries. If the ICS site is maintaining guide documents, then another task includes entering guide URLs in the collection descriptors to relate guides to collections. The *CMT* is used in conjunction with the data base administration tools which the *Retrieval Manager* provides.

The *Monitoring and Control Tools (MCT)* (Figure 3-10), provide the machine-to-machine interface for integrating the operations of the *Retrieval Manager* with the operations of a site. Through the *MCT*, the monitor and control operations of the *Retrieval Manager* become part of the *SSM* operations for the data provider. (The *MCT* has been identified as an element in the ICS but specific functionality will be detailed in future releases of the SDD.)

ISA

Figure 3-10. Other ICS Elements

3.3 Catalogue Interoperability Protocol (CIP) Domain Design

The Catalogue Interoperability Protocol (CIP) is the main method by which the objective of catalogue interoperability is achieved in ICS. CIP is the main interface method between ICS elements. CIP is elaborated in the CIP Specification [R3]. The CIP Specification defines the services and attributes which are specific to EO data catalogues and common across *Retrieval Managers* and *Translators* which speak CIP.

3.3.1 CIP System Design

The following sections discuss configurations of CIP elements from the perspectives of conformance to the protocol, interoperability with existing systems, and discovery of data within ICS.

3.3.1.1 Maximum ICS CIP Site

ICS Compatability: Explanatory

This section defines how the ICS elements are configured into a data provider site that needs search capability for collections and products. The elements were defined in Section 3.2. The site described here is a maximum site in that it contains all ICS elements (Figure 3-11) from the CIP domain. Particular sites have subsets of the elements. Variety amongst sites is addressed in the next section. The top half of the figure shows those elements which are strictly part of ICS. The bottom half of the figure shows those elements which exist within the *CEOS Agency Systems* and for the purposes of this discussion are labeled as ICS Related Elements.

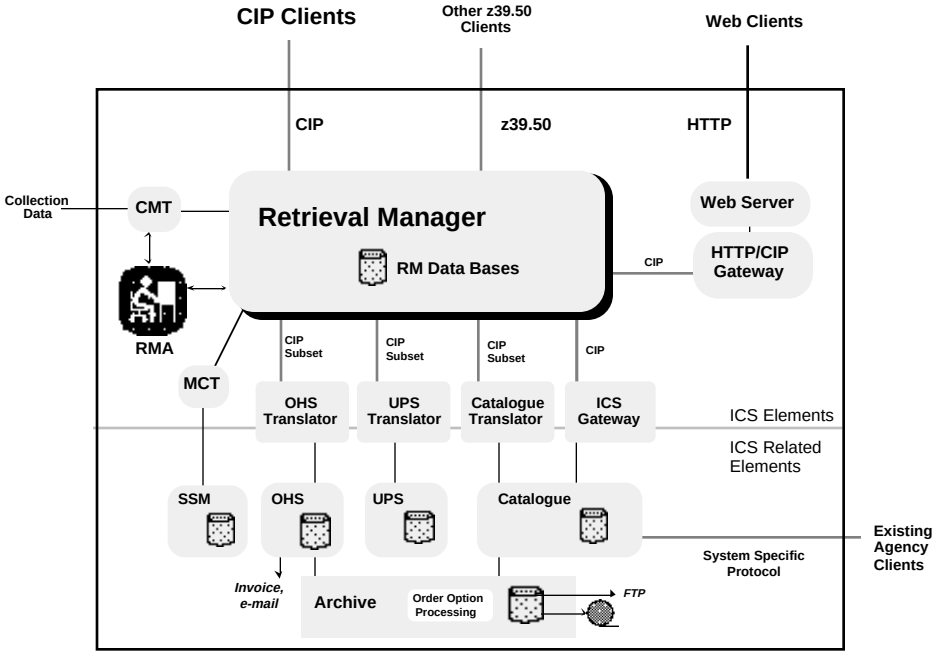

Figure 3-11. Maximum ICS CIP Site

The heart of an ICS site is the *Retrieval Manager*. Virtually all CIP communication through the site passes through the *Retrieval Manager.* A variety of Clients are shown interacting with the *Retrieval Manager.* A connection to an *ICS Client* is shown. WWW access is provided through a Web Server and *HTTP/CIP Gateway* interacting with the *Retrieval Manager*. The *Other Z39.50 Clients* are those clients which communicate using a profile of Z39.50, Version 3, but not CIP. These clients will be able to access some but generally not all of the *Retrieval Manager* services. An additional client is also shown in Figure 3-11, *Existing Agency Clients* will continue to access the site's catalogue. The site *catalogue* provides access to ICS through an *ICS Gateway*. This allows users of existing agency clients to perform searches and ordering of ICS items.

The *Retrieval Manager* interacts with a series of *Translators* to access the data providers services. Figure 3-11 shows one appraoch to translator configuration at a site (See Section 3.2.2.3 for other approachs). Between the *Retrieval Manager* and the *Translators*, a subset of CIP messages particular to the *translator* are used for communications. Only a subset is supported because the *Retrieval Manager* routes queries to the *Catalogue* and orders to the *OHS*, therefore an *OHS Translator* is not designed to respond to queries. Between the *Translators* and the provider site services, the protocols for communication within the site are used. The data provider elements perform the necessary catalogue services or interact with other site resources, e.g., archive, to respond to the requests coming from the *Translators*.

Several other elements are shown in Figure 3-11 which pertain to keeping the ICS site operational. The *ISA* is shown interacting with the *Retrieval Manager* to establish and maintain the *Retrieval Manager*. This is done through a Graphical User Interface (GUI) provided by the *Retrieval Manager*. The *ISA* uses the *CMT* to populate and maintain the data in the *Retrieval Manager*. To support integration of *Retrieval Manager* operations with the *SSM*, an *MCT* is provided to monitor an control *Retrieval Manager* operations as directed by the *SSM*.

3.3.1.2 ICS as a System

ICS Compatability: Explanatory

Figure 3-12 shows the ICS Functional Framework. The framework shows a variety of arrangement of elements at ICS sites. Specific arrangement of ICS elements provided at ICS sites will vary over time. That is Clients, *Retrieval Managers* and even Agency holdings will be added, or taken away, dynamically, during the ICS operations. The ICS must enable this dynamic configuration. Therefore the ICS Framework shown in Figure 3-3 is only representative of how the ICS will be implemented at any given time.

The ICS Framework shows four sites which emphasize different roles at the sites. The four roles are described in this section: *Retrieval Manager* with an Included Catalogue, *Retrieval Manager* as a gateway to Multiple Catalogues, *Retrieval Manager* with only Theme Collections, and *Retrieval Manager* as a Gateway to an Agency Data System.

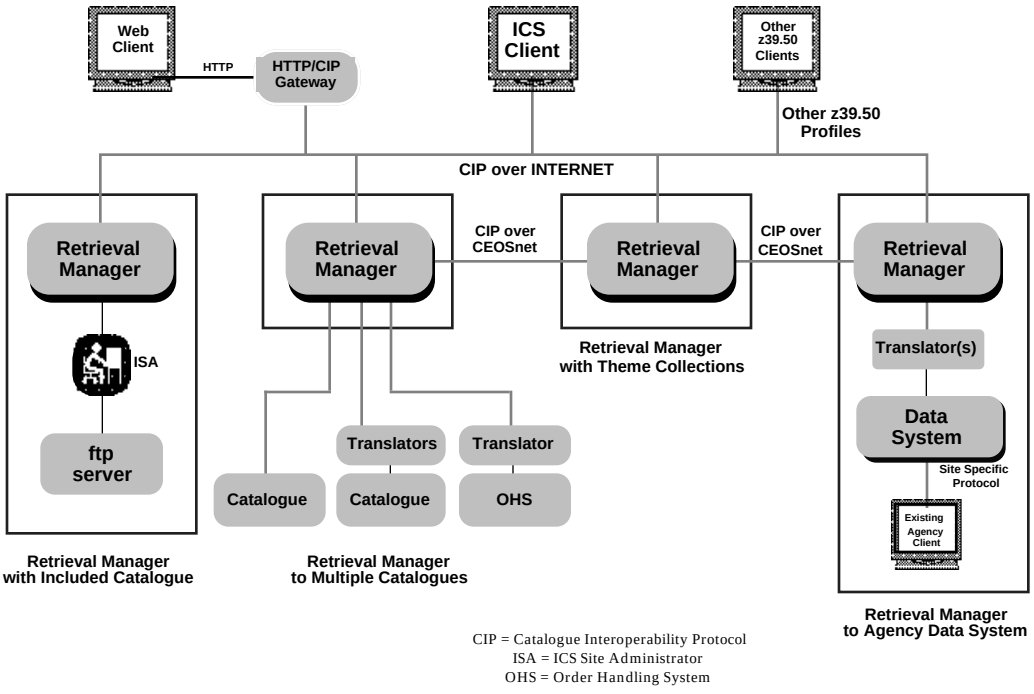

Figure 3-12. ICS CIP Functional Framework

Retrieval Manager with an Included Catalogue. The middle configuration in Figure 3-12 shows how a *Retrieval Manager* is used to provide access to a more limited archive with minimal resources. In this case, the *Retrieval Manager* is a standalone system which provides the *collections* and *product* searches. No *Translator*s are used. The *Retrieval Manager* contains the metadata for *collections* and *products*. The user then retrieves the actual *products* via ftp, for example, from the archive. Ordering and authentication need not be implemented at this site. Setting up this form of a *Retrieval Manager* may be simplified by implementing the *Retrieval Manager* databases as a set of ASCII indexed files. Administration of the indexed files is accomplished through the *Retrieval Manager*.

Retrieval Manager as a gateway to Multiple Catalogues. A *Retrieval Manager* may be used as a gateway to multiple CIP accessible catalogues. The *Retrieval Manager* serves as the access point for a cluster of catalogues. The cluster of catalogs for the *Retrieval Manager* are considered local to that *Retrieval Manager* even though they may be geographically distant. The *Retrieval Manager* holds the collections which include products from the local catalogues. The *Retrieval Manager* may also reference collections in other *Retrieval Managers*, i.e., remote collections. Collection searches are performed in the *Retrieval Managers*. Product Searches are forwarded to the catalogues in the cluster.

Retrieval Manager with only Theme Collections. An additional variant of the *Retrieval Manager* implementation is an ICS router node in the ICS Framework. A *Retrieval Manager* as a router functions as a "well known" ICS resource enabling users to find other *Retrieval Manager*s for specific data providers. An ICS Router is a middleware node containing *collections* but no *product* metadata. For Release B, an ICS router will be used for the ICS Global Node providing a "well known" resource to access all ICS *Retrieval Managers* (see Section 3.3.1.4 for a discussion of the ICS Global Node).

Retrieval Manager as a Gateway to an Agency Data System. For Agencies with existing catalogue and archive systems, the *Retrieval Manager* holds the *collections* which are particular to the ICS domain. The *product* metadata which is in the agency catalogue database and is accessed through a *Catalogue Translator* dynamically to satisfy queries. Likewise, the *Retrieval Manager* in this case holds user management data for only the ICS domain users (which will include the other *Retrieval Managers*). The *Retrieval Manager* interfaces to existing *User Profile Systems* and *Order Handling Systems* through *Translators*.

Multiple configurations for *Retrieval Manager* -to-*Translator* interfaces are supported. As shown in Figure 3-12, a *Retrieval Manager* can direct CIP messages to more than one*Translator*. The converse is not true, namely a *translator* can not be served by more than one *Retrieval Manager*. Limiting the *Translator* in this fashion reduces the burden on the *translator* to manage CIP messages from more than one source, making the *translator* a simpler design. The purpose

of the multiple transaltors for a single *Retrieval Manager* allows clusters for*catalogues* to be served by a single *Retrieval Manager*. The multiple catalogues may be at different sites but under the same administration. Or a site may host a *translator* and *catalogue* and rely on a different site and organization to manage the *Retrieval Manager.* If a *catalogue* needs to provide access to multiple *Retrieval Managers,* multiple *translators* are provided as the existing *catalogue* is designed to handle multiple clients.

3.3.1.3 CIP/GEO Interoperability

This section describes the approach to supporting interoperation between the CIP domain and GEO - another Z39.50 based protocol.

3.3.1.3.1 Overview

ICS Compatability: Explanatory

One reason for basing CIP on Z39.50 is the compatibility with other Z39.50 based systems. The GEO Profile [R25] is the basis for several systems which provide access to geospatial metadata conforming to the US Federal Geographic Data Committee (FGDC) standard. CIP - Release B and the GEO profile have been aligned to support interoperability on a limited basis between GEO clients and CIP servers and between CIP clients and GEO servers. This alignment increases the data accessible to both GEO and CIP users.

There are differences between the GEO and CIP models. GEO is a 2-Tier, client-server profile. CIP extends the Z39.50 protocol to a 3-Tier, distributed model (see Figure 3-11). The GEO data model consists of datasets held at GEO servers. GEO datasets correspond to CIP Products. CIP Products are aggregated into Collections. CIP allows aggregation of Collections to support Theme Collections and to support distributed searching. Due to these differences in architectures and data models, there must be constraints on which collections/servers are targeted when crossing the CIP/GEO domains. These constraints are discussed in the following sections.

References to CIP Collections in GEO systems is done using a CIP collection URL as defined in [R3]. References to GEO servers in CIP Retrieval Managers is done using a URL as defined in [R25]. A GEO data administrator is responsible for maintaining the directory information of CIP collections in their system. The ISA is responsible for maintaining the collection database entries for GEO targets listed in the ISA's Retrieval Manager.

3.3.1.3.2 GEO to CIP Interoperability

ICS Compatability: Explanatory

To make CIP data available to GEO users, ICS must provide GEO based systems access to Retrieval Managers. To support access to distributed GEO servers, several GEO based systems maintain some form of directory, which can be referred to as a Directory Of GEO Server

(DOGS). DOGS serve as an access point for a client to send a single search which will be directed to one or more GEO servers (See Figure 3-13). A DOGS fetches the results from the multiple GEO servers and return a single response to the user's client. For CIP Interoperability, Provider Archive Collections are listed in the DOGS. (Note that Theme Collections cannot be listed in the DOGS as the GEO data model does not allow collections of collections.) The DOGS may allow for discovery of the collections within the DOGS and may also allow for searches to be propagated from the DOGS to the GEO servers.

When a DOGS determines that a GEO search should be targeted at a CIP Provider Archive Collection, a Z39.50 Version 2 session is established with the Retrieval Manager which provides access to the collection and a search request using the GEO attribute set is sent. Because the GEO and CIP attribute sets have been partially aligned the Retrieval Manager is able to pass the search request to the appropriate catalogue as a CIP local product search request. (For details on the behavior of non-aligned attributes, see the CIP Specification [R3].)

The arrangement of elements in the GEO world allows distributed searching using a 2-Tier protocol. Distributed searching is accomplished using two, 2-Tier client-server interactions with the DOGS in the middle. The DOGS accept requests from a user's client via an application protocol, e.g., http. Using the directory of servers held in the DOGS, the DOGS forms Z39.50 sessions with one or more servers using the GEO profile. Each session is a client-server session with a GEO server. The DOGS are the key item in these approaches to distributed searching.

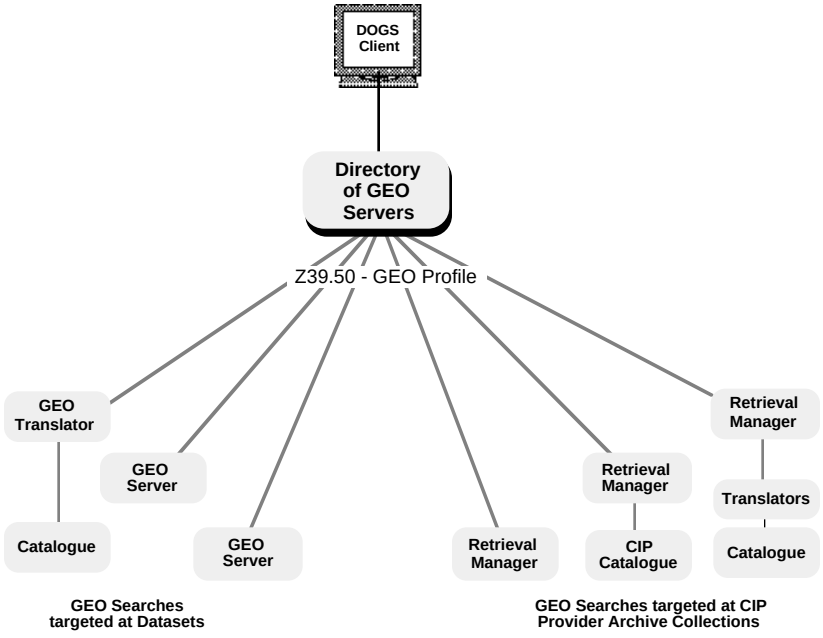

Figure 3-13. CIP Interoperability with GEO

3.3.1.3.3 CIP to GEO Interoperability

ICS Compatability: Explanatory

To make GEO data available to CIP users, GEO based systems must provide access to GEO servers for ICS *Retrieval Managers*. GEO servers provide search information retrieval services on GEO datasets. The GEO datasets are listed in a *Retrieval Manager* collection database as a product in a collection. These "GEO collections" can then be targeted for searching either directly or through inclusion into Theme collections. When a user targets a product search at a collection which contains a GEO dataset, the *Retrieval Manager* will establish a Z39.50 Version 2 session with the GEO target. The product search will be sent using the CIP attribute set.

The interoperability approach allows mirroring a DOGS registry as a collection of product archives within a *Retrieval Manager* as shown in Figure 3-14. There is the potential here to build a richer collection structure of GEO servers than is currently available through a DOGS registry. GEO servers could be grouped into thematic sub-collections. This structure would only be visible to true *ICS Clients* via collection hierarchies. A *ICS Client* would submit a product search to a selected set of GEO servers via the collection hierarchy.

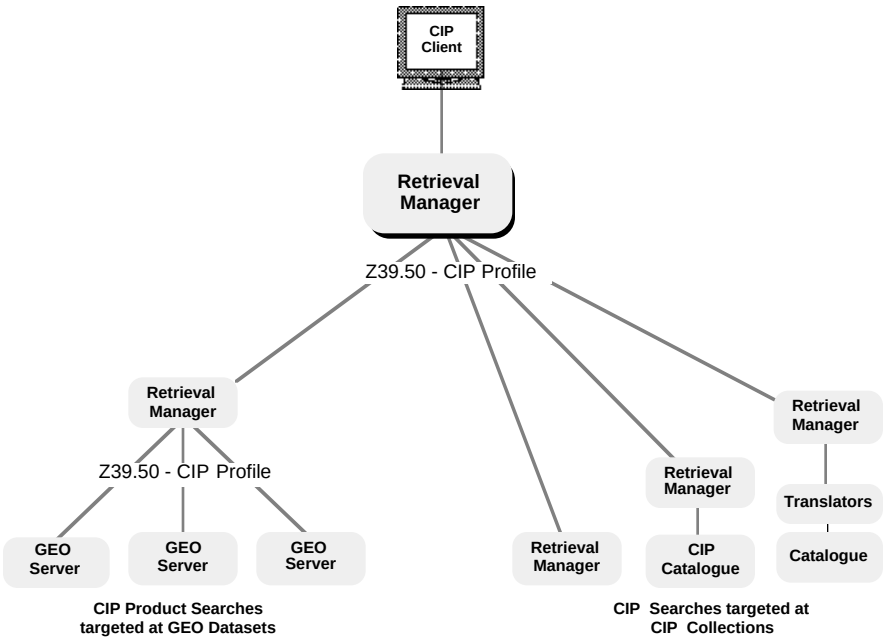

Figure 3-14. GEO Interoperability with CIP

3.3.1.4 Global Node Site

ICS Compatability: Explanatory

This section defines the concept of an ICS Global Node site. Because ICS provides the framework to link many different data providers, it becomes increasingly important to provide a convenient way for users to *discover* the kinds of data (i.e., products and documents) that is available to them within the system. An ICS Global Node provides the ability to the user to discover collections across the system.

An ICS Global Node maintains information about each root node that it can access. The root node is defined by the root collection descriptor maintained in the Collection Database managed by a given RM. A Global Node Administrator maintains a small text description of each accessible root node along with the URL of the root node CDB so that the Global Node knows how to find it.

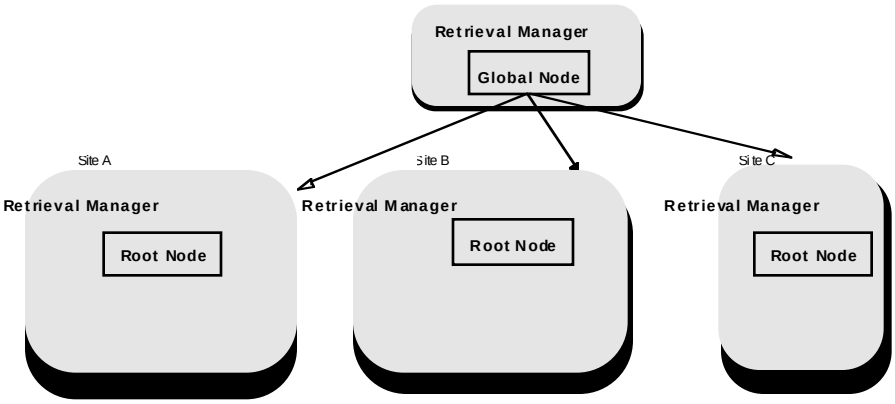

Figure 3-15. Global Node Site

Via an ICS Client, the user can perform a search against the Global Node to see a list of the available root nodes along with the text description. To view the collection descriptors for each root node, the Global Node issues a search request to each of the RMs to which it can connect. Should the user desire to see descriptors for collections below the root collection descriptors, a more refined collection search can be posed through the ICS Client. As shown in Figure 3-15, the Global Node again issues the search request to each of the RMs to which it can connect. In this case, each RM performs a “local” search on its collections. A collection descriptor is returned by the RM for each local collection that satisfies the given search criteria.

It is important to understand the local collection search constraint that must be implemented by the RM. For example, if the search criteria selects a theme collection which is comprised of local collections and one or more remote collections, the given RM returns only the collection descriptors for the local collections, assuming that they meet the search criteria. The reason for this is that the Global Node has issued the search request to all RMs. If the "remote" collections that comprise part of the theme collection satisfy the search criteria, then their collection descriptors will be returned by the RM where they are stored. This constraint prevents the problem of redundant searches and thus, redundant collection descriptors in the Global Node foundset.

From an architecture perspective, Global Nodes can be replicated within ICS. This permits distant sites to more easily perform discovery searches. Global Nodes themselves are not complete instances of RMs; they have a much pared down architecture. And, it is assumed that once a user decides to issue a product search based on the collection descriptor information retrieved by the Global Node, that the ICS Client then switches from a Global Node connection to a Retrieval Manager connection. In other words, the Global Node does not act as a "pass-thru" mechanism for product searches.

3.3.2 CIP Protocol Overview

ICS Compatability: Explanatory (See R3 for CIP Compliance)

CIP messages are exchanged between an *origin* and a *target* as part of a *CIP Session*. An *origin* initiates a session with a *target*. Several ICS elements act as *origins*: *ICS Client, Retrieval Manager, ICS Gateway, and HTTP/CIP Gateway*. Several ICS elements act as *targets*: *Retrieval Managers* and *Translators.*

The messages which CIP uses are listed in Table 3-1. The messages are used in the SDD to demonstrated the use of CIP in ICS. The CIP Specification [R3] provides a controlled definition of the contents and meaning of each of the messages in Table 3-1.

Table 3-1. CIP Messages

CIP Message Name	Short Description of CIP Message
InitializeRequest	Used by an *origin* to establish a *session* with a *target*.
InitializeResponse	Generated by the *target* after reception of the *InitializeRequest*
SearchRequest	Defines the elements transferred for any search operation including the query, the target database, and a result set name
SearchResponse	Contains the response to a *SearchRequest*, in particular it contains the total number of records in the result set
PresentRequest	Specifies the characteristics of the records in a result set which are to be returned to the *origin.* Included in the *PresentRequest* is the range of records to be returned.
PresentResponse	Return of the records requested during a *PresentRequest*.
SegmentRequest	When the records requested by a *PresentRequest* will not fit in a single segment , the *target* returns multiple *SegmentRequest*, each containing a portion of the requested records.
DeleteResultSetRequest	Allows a *target* to request an *origin* to delete specified result sets created during a *CIP session*
DeleteResultSetResponse	Generated by the *target* in response to the *DeleteResultSetResponse*
AccessControlRequest	Allows a *target* to challenge the identity of an *origin* .
AccessControlResponse	Sent by the *origin* to the *target* in response to a "challenge" to the identity of the *origin* through an AccessControlRequest
ResourceControlRequest	Allows the *target* to provide information concerning the status of a query to the *origin*.
ResourceControlResponse	Allows the *origin* to indicate whether to continue a query or not.
TriggerResourceControlRequest	Used by the *origin* to request a status report or change the status of an existing query. May be used to cancel a query started by the *origin* .
ResourceReportRequest	Allows an *origin* to request a *target* to produce a resource report.
ResourceReportResponse	Generated by the *target* in response to the *ResourceReportRequest* .
ExtendedServicesRequest	Allows an *origin* to perform tasks which are executed outside the scope of Z39.50. CIP supports the following extended services: Persistent Result Sets, Persistent Query, Periodic Query Schedule , Database Update, CIP Ordering The CIP Ordering Extended Service can contain one of the following actions: *orderQuoteAndValidate, orderSubmit, orderMonitor,* and *orderCancel.*
ExtendedServicesResponse	Allows a *target* to send back the response related to the execution of tasks executed outside the scope of Z39.50 and initiated by an *ExtendedServicesResponse.*
Close	Allows an *origin* or *target* to abruptly terminate all active operations and to initiate the termination of the *CIP Session*

Session Initialization and Closure. CIP is a session based protocol. A *CIP session* begins with an initialization *operation* and ends with a closure *operation*. The initialization allows negotiation procedures to control the use of all services to be used during the session. The closure *operation* signals the end of the session and is done unilaterally by either the *origin* or *target*.

Search. The Search service is the means by which the *origin* sends a query to be evaluated against the EO data and the status of the query is returned. A CIP search consists of a number of fundamental parts:

- Search control parameters
- The search query using a Z39.50 language and containing a valid combination of CIP attributes
- The target database for the search
- The result set to be created at the *target* to hold the results of the search
- Information about the structure of a distributed result set

Search Control. Search control parameters allow the user to control the type of search requested. The user can select either a *collection* search or a *product* search. A *collection* search is a CIP search that is used to identify and retrieve *collection* definitions. A *product* descriptor search is a CIP search that is used to identify and retrieve *product* descriptors. Also, the user can select any search to be either local to the *Retrieval Manager* which holds the targeted *collection*, or allow the search to be distributed to other *Retrieval Manager*s as identified in the searched *collection*.

Query Language. CIP defines queries using a general purpose query language defined using Reverse-Polish-Notation (RPN). Note, however, that this does not preclude *Retrieval Managers* from supporting other additional Z39.50 query languages.

Search Attributes. CIP splits attributes into different types depending upon their function: Use, Relation, Position, Structure, Truncation and Completeness attributes. In a search query, the Use attribute identifies the access point against which the search term is to be matched. A set of Use attributes based on the EO domain are defined and controlled in the CIP Specification. The other types of attributes are used to provide additional match criteria in a query.

In addition to the CIP attributes defined in the CIP specification, each data provider may define its own local attributes. These local attributes are used in exactly the same way as the CIP attributes. The only restriction is that their understanding is limited to the domain of the data provider who defines them (instead of the whole CIP domain as for the CIP attributes). For instance, if a search query contains a local attribute, this local attribute will be applicable (i.e. will be recognized) only by the *collections* owned by the appropriate *Retrieval Manager*.

Result Set Information. As a distributed CIP search may result in results at multiple targets, the *searchResponse* contains a description of the distributed result set, e.g., the number of records in each of the lowest level result sets. With this information, the *origin*, can request records from specific result sets using a subsequent *presentRequest.*

Result Set Retrieval. Using a *presentRequest*, CIP returns records that have been located through execution of the search service. Once a CIP *searchRequest* has been submitted to the *Retrieval Manager*, performed at the *target*(s), and has responded successfully, a result set is

made available for subsequent *presentRequests* by the *origin*. Optionally, an *origin* may request piggybacking in a CIP search request, in which case a small number of retrieval records are returned in the search response itself. The retrieval record formats for piggybacked records are the same as those described for present responses. By selecting from pre-defined element sets, the user can select the desired fields to be returned from the result set.

Extended Services. *Extended Services* provide a mechanism to define and monitor tasks that are executed outside Z39.50. An *Extended Service* (ES) defines a particular task which is related to information retrieval but is not defined within Z39.50. It allows an *origin* to create, modify or delete *task packages*, which are maintained by the *target* in a database - the *Extended Services database*. The task defined in a *task package* depends on the particular *Extended Service* used. For CIP the following *Extended Services* are provided: Persistent Result Sets, Persistent Query, Periodic Query Schedule, Database Update, and CIP Ordering. Note that an *ES response* does not necessarily signal the completion of a task, which may have a lifetime which exceeds the *CIP Session* during which it is initiated.

Ordering. CIP defines an *Extended Service* for ordering of EO *products*. CIP supports submitting an order specification for validation and to obtain a quote. A specific message is provided for affirmative submission of an order. Additionally, a user can monitor the status of a order and request to cancel the order. Options on an order are specified in the order specification of CIP order message. Details on order options can be found in the Order Option Ammendment to the CIP Specification [R28].

Authentication: CIP provides the capability to transfer authentication information between *target* and *origin* pairs of a *CIP Session*. This transfer allows *Retrieval Managers* to authenticate *ICS Clients* and other *Retrieval Managers*. The authentication may be done at the beginning of a *CIP Session* or may be requested by a *Retrieval Manager* for a specific *operation*. Additionally, a *Retrieval Manager* can request non-repudiation be enforced for a specific request.

Explain. The Explain service enables the capabilities of a target server to be ascertained by clients. Explain queries are targeted at the Explain database of a *Retrieval Manager*. The Explain database contains the attributes which can be used at the *Retrieval Manager* as well as key information about the EO content of the *Retrieval Manager*. The Explain database also provides information about the format of records returned to the *origin* by the *Retrieval Manager*.

Identifiers. The CIP Specification also defines syntax for identifiers of several key items in the CIP space: messages and *collections* (including *task packages*). For example, each operation request from a client (or *Retrieval Manager*) must have a unique reference identifier. This can be used in the tracking of an operation and is particularly important when a search is invoked which is passed onto remote *Retrieval Manager*s.

Unsupported Operations. Not all ICS elements will support all CIP operations. Negotiation of supported operations occurs during *session* initialization. ICS elements as part of initialization will identify the operations supported by an element. If a message for an unsupported operations is sent during the session, e.g., a CIP ordering message is sent to a Catalogue Translator, the ICS element shall respond to the unsupported message with a diagnostic message.

3.3.3 CIP Operations

Messages are passed between ICS elements primarily using CIP. CIP is a session based protocol for conducting the operations, e.g., search and order. This section shows how various messages are passed between the distributed ICS elements to accomplish a request by a user for a query or an order. Each section discusses the simpler "local" case, and then discusses the case when multiple *Retrieval Managers* are involved. The operations shown here are examples of the two main interactions in ICS - search and order - and do not demonstrate the full extent of how ICS is used.

In each scenario it is assumed that a session is begun before the scenario is described. A scenario is established by the Client sending an *InitializeRequest* to the *Retrieval Manager* and the *Retrieval Manager* replying with an *InitializeResponse*. Likewise when a *Retrieval Manager* must send a secondary message to a another *Retrieval Manager* or to a *Translator*, a session is formed between the two. Nominally, the Client will send a *close* message to end the session. There is no response from the *Retrieval Manager* to the Client for a *close*. When the *Retrieval Manager* receives a *close* it will send a *close* to any other ICS elements to which a session had been established for this user.

3.3.3.1 Queries

When the user wants to find records pertaining to a particular interest, there will be multiple CIP messages behind the scenes which the ICS Client may or may not make visible. The CIP messages provide the flexibility to retrieve only the information that the user wishes to view. For instance a query consists of two separate operations: a search operation and a present operation. The search operation returns the number of hits. The present returns the records in the particular format requested by the user.

The following two sections show how the combination of search and present operations are used to search ICS data. First a local query is described followed by a distributed query. The particular examples shown in this section are typical but not the only combination of operations which a user may cause to happen in ICS. For example, a user query session may include multiple searches and only one present.

3.3.3.1.1 Local Query

ICS Compatability: Mandatory

Figure 3-16 shows the passing of messages between ICS elements to accomplish a local query. The query is searching the *collections* held by a local catalogue. The ICS elements involved are an *ICS Client Application*, a *Retrieval Manager*, and a *Catalogue Translator*. The client targets a *searchRequest* at a *collection* held by the *Retrieval Manager*. The contents of the *collection* in this case are held in the local catalogue and so the *Retrieval Manager* passes a *searchRequest* to the *Catalogue Translator*. The *searchResponse* which is returned from the Catalogue *Translator* and is passed to the client contains the number of "hits" produced by the search.

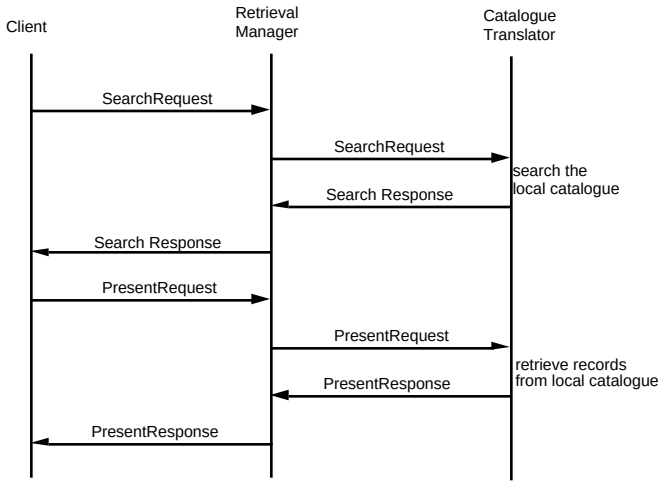

Figure 3-16. Local Query Operation Messages

Assuming that the user is satisfied by the number of hits indicated, the *ICS Client* is used to request the records in a specific format. This is accomplished by the client issuing a *presentRequest* against the result set which is held by the *Catalogue Translator*. The records are retrieved from the local catalogues and passed to the client via the *Retrieval Manager.*

The main function of the *Retrieval Manager* in this case has been to route messages to appropriate ICS elements based on the contents of the *collections*. Messages are routed based on what type of search was requested and where specific data is held. A user can request that a CIP search be either local to the *Retrieval Manager* or be propagated to other *Retrieval Managers* if the collection structure would warrant this for a specific query. The user may also choose between searching just the collections or searching the included products as well. If an ICS site holds the product descriptors in the local catalogue and not in the *Retrieval Manager*, the search will need to be forwarded to the catalogue if a Product Search was requested. How data should

be distributed between the *Retrieval Manager* and the catalogue is dependent upon characteristics of an agency's data, e.g. number of products and volatility of *collections*. Each agency will determine the distribution of data while considering the aspects of maintaining replicated data versus the response time for processing a user request. These issues can be assessed at the time of configuring an ICS site. The CEOS PTT plans to provide guidance on these issues in the ICS Collections Manual [R5].

The scenarios for the *Navigate* and *Locate* services are done similar to the Search scenario above. When a user knows a URL for an item, e.g., a collection, the user can locate that item by sending a search to the *Retrieval Manager* indicated in the URL where a search is performed, e.g. against the collection database. Once a successful search has been completed, a present request can be made to retrieve the item descriptor. Similarly for the navigation service, it is assumed that user has an item descriptor at the ICS Client and wishes to change context to a related collection. Where the related collection is listed as part of the collection descriptor. To retrieve the related collection, a search for the collection's URL must first be performed successfully, followed by a present request and response.

Any CIP search operation that is performed to achieve a locate or navigate scenario is done as part of a CIP session. In order to minimize the overhead of session handling, the CIP Client checks if a session is already established with the Retrieval Manager to which the seachRequest will be sent. If a session already exists with the Retrieval Manager, the Client sends the searchRequest as part of that session otherwise a new session is established.

3.3.3.1.2 Distributed Query

ICS Compatability: Mandatory

The distributed query capabilities of the ICS are depicted by the messages in Figure 3-17. Similar to the previous section, the following operations are handled in a *CIP session* which is assumed to be established prior to the *operations* shown here.

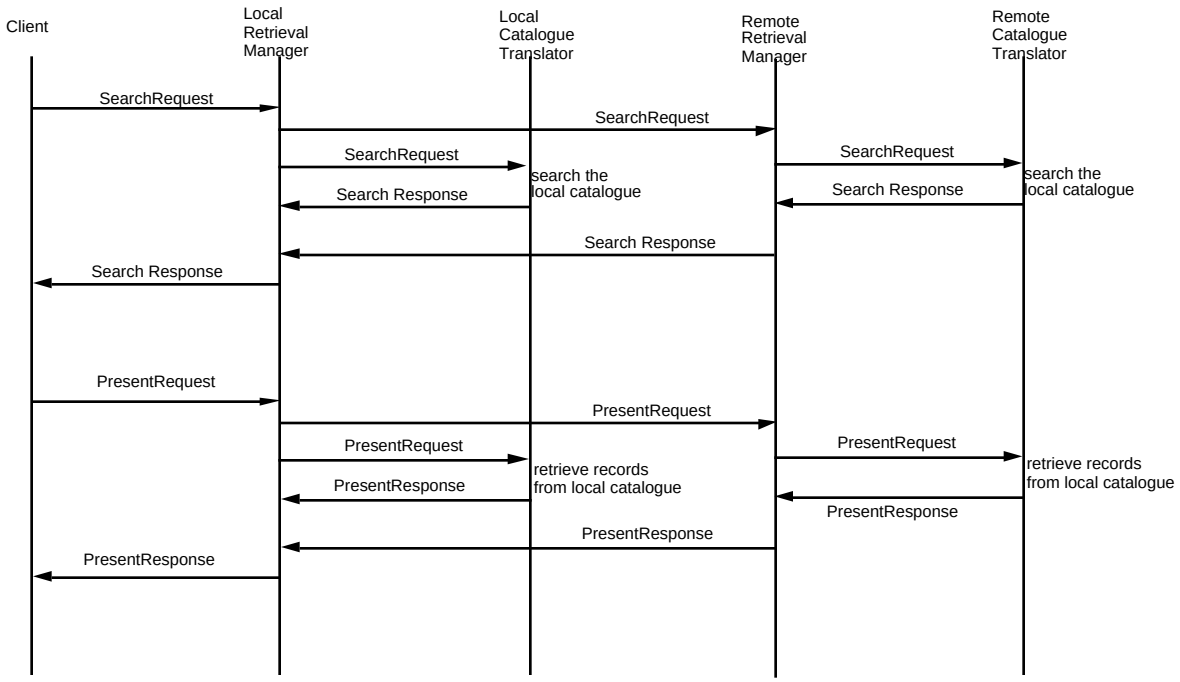

Figure 3-17. Distributed Query Operation Messages

When an*ICS Client* targets a *searchRequest* at a *collection* which includes both local *collections* and remote *collections*, the *Retrieval Manager* creates sub-searches which are targeted at the included *collections*. In this particular example, two separate *searchRequests* are created: one is sent to the local *Catalogue Translator*, the second is sent to a remote *Retrieval Manager*. The remote *Retrieval Manager* in turn determines, based on the *collection* contents, that a *searchRequest* needs to be sent to a *Catalogue Translator* which is local to the remote *Retrieval Manager*. *SearchResponses* are routed back to the initial *Retrieval Manager* which combines them, and a single *searchResponse* is sent to the client. The *searchResponse* contains information about the structure of distributed result sets, e.g., the number of records in each leaf level result set.

Similarto the splitting of a *searchRequest*, a *presentRequest* is split into sub-requests based on the contents of the result set. The *presentRequests* are routed to the distributed result sets and *presentResponses* are returned to the initial *Retrieval Manager.* A *presentRequest* can be directed to return all records from all of the distributed result sets. Because the sub-requests could each bring back large result sets, CIP provides the ability for an *ICS Client* to request records from specific distributed leaf result sets. An *ICS Client* composes such a detailed *presentRequest* using the description of the distributed result set found in the *searchResponse*.

An *ICS Client* can keep the user informed of the status of a search using the CIP Resource Control Facility. When in effect, Resource Control provides the structure for the *Retrieval Manager* and *ICS Client* to exchange messages about the status of an active operation, e.g., a distributed product search. Resource Control can be in effect for an entire *CIP session* or an *ICS Client* can trigger the Retrieval Manager to send a report for a specific operation only. If during initialization, Resource Control was negotiated to be in effect for the session, the *Retrieval Manager* will send *resourceControlRequests* to the *ICS Client* while an operation is active. The *resourceControlRequest* will provide a resource report including the status of the operation and may allow the client to stop the operation using a *ResourceControlResponse.* If resource control is not in effect for the session, the *ICS Client* can send a *TriggerResourceControlRequest* to which the *Retrieval Manager* will respond with a *ResourceControlRequest* including the resource report with status. For a distributed operation, the resource report contains information about sub-requests. This allows a user to determine, for example, that several of the distributed product sub-searches have been completed but the entire operation has not completed because of a single translator being down.

Users also want to know if a portion of a distributed seach has failed. When a sub-search returns with a failed status, the *Retrieval Manager* allows the other sub-searches to go to completion. If Resource Control is in effect for the session, the failure of a sub-search will be included in the *ResourceControlRequest* which is sent to the client. When all of the sub-searches are completed, a *searchResponse* is returned to the *ICS Client.*

After doing a distributed collection search, a user may target a product search at a specific remote collection. The user's client may need to access information about the collection in order to do an accurate query. The client does a *searchRequest* of the Explain database. The client makes this request against the local *Retrieval Manager*, which forwards the searchRequest on to the remote *Retrieval Manager* which holds the collection. The remote *Retrieval Manager* replies back to the local *Retrieval Manager* with the *searchResponse.* The local *Retrieval Manager* sends the *searchResponse* to the *Client.*

3.3.3.1.3 Global Node Queries

ICS Compatability: Mandatory

The global node query capability of the ICS is depicted by the messages in Figure 3-18. Like the previous sections, the following operations are handled in a *CIP session* which is assumed to be established prior to the *operations* shown here.

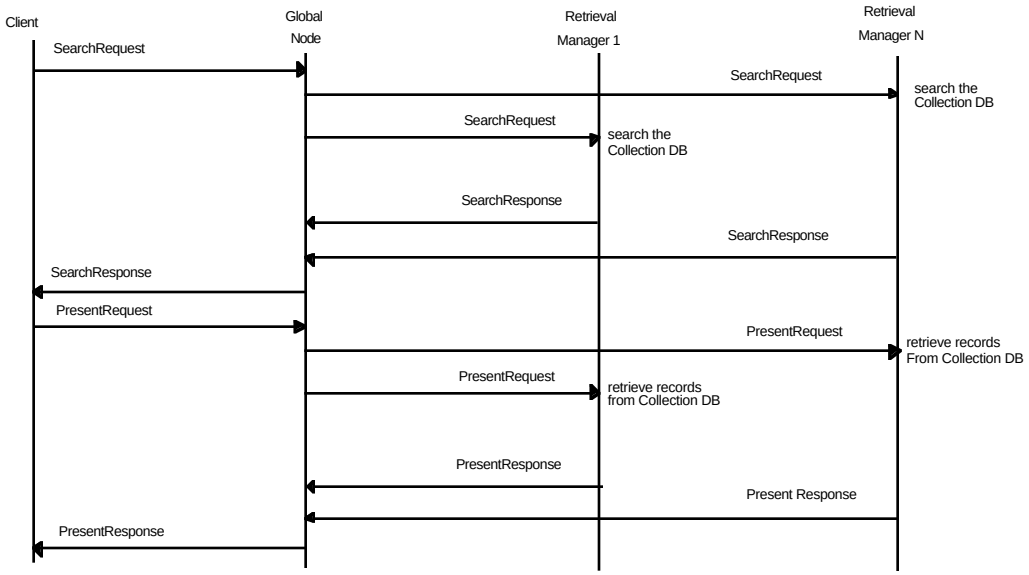

Figure 3-18. Global Node Query Operation Messages

When the *ICS Client* targets a *searchRequest* at a *Global Node*, the *Global Node* issues a *searchRequest* to each of the *Retrieval Managers* to which it can connect in order to search the Collection Database. The *searchRequest* parameter *additionalSearchInfo* indicates that the search is to be performed only locally. That is, if a theme collection is found that points to a remote collection, only the matching local component collections are returned as part of the *searchResponse.* Each *Retrieval Manager* then returns the *searchResponse* to the *Global Node.* The *Global Node* aggregates the *searchResponses* and returns the composite *searchResponse* to the *ICS Client.*

The handling of the *presentRequest* and subsequent *presentResponses* occurs in the same way as the *searchRequest* and *searchResponses* for a Global Node Query. The composite *presentResponse* contains the collection descriptors for all of the ICS collections that match the query criteria. This enables the user to "discover" collections across the entire ICS.

3.3.3.2 Ordering of EO Products

This section describes ordering of EO Products using CIP. The ordering process is shown here as fundamentally a two step process. First, a step of submitting an order specification resulting in validation and a quote. Second, the order is submitted for processing. There are multiple variations on this two step process. There may be multiple quotes requested before the user is happy with the order specification and then the order is submitted. On the other extreme the user may know exactly what they want and will directly submit the order without first asking for

a quote. The above two step process assumes that the order items, i.e. the products which must be ordered, have already been identified. This is performed by querying collections and performing refinements on the queries until the desired order items and their related order options have been identified, by using the *Search* and *Present* facilities described in the previous section. A user can also monitor and cancel an order, but these operations are not shown in the following scenarios.

This section, like the last previous section, first shows a single *Retrieval Manager* case followed by the case in which two *Retrieval Manager*s are involved. There is a key difference between queries and orders. A query may be distributed to multiple *Retrieval Manager*s and *Catalogues*. Orders must contain *products* from a single *OHS*. When the order is composed by the client it will contain requests for items from only one site. The order may be routed from the local *Retrieval Manager* to a remote *Retrieval Manager*, but the order will only be presented to one *OHS*. If products need to be ordered from multiple OHS', several order requests must be submitted. For this reason we distinguish between "distributed" queries vs. "indirect" ordering.

Critical to ordering for some data providers will be the security mechanisms provided by CIP. Data providers will restrict some *products* to particular individuals and a monetary charge will be required for some *products*. CIP provides an authentication mechanism allowing a data provider to verify the identity of the person requesting a product. CIP also provides non-repudiation, which when invoked requires the user to submit an order which can not be disavowed at a later time. A detailed scenario using CIP authentication is provided in Section 6.

3.3.3.2.1 Direct Ordering

ICS Compatability: MAA

The direct ordering scenario by a user involves a *primary order*, from the *ICS Client* to the *Retrieval Manager*, and a *secondary order* to the local OHS *Translator* (See Figure 3-19). A *primary order* is defined to be between an *ICS Client* and a *Retrieval Manager*. A *secondary order* is created by an *Retrieval Manager* in response to a *primary order* and may be either to another *Retrieval Manager* or to an *OHS Translator*. Like in the previous scenarios, the messages shown occur as part of a *CIP Session* for which the initialization *operation* is not shown in the scenario.

After the user has specified the contents of the order, the client sends a CIP Ordering *extendedServicesRequest*, with the action of *orderQuoteAndValidate*, to the *Retrieval Manager*. To support the ordering *Extended Service*, the *Retrieval Manager* creates a *Task Package* which contains the order as well as additional information related to the *operation*. The *Task Package* will be updated based on the additional *Extended Services* messages as the scenario proceeds. The *Retrieval Manager* determines which OHS can fulfill the specified order. In this case, the

extendedServicesRequest is routed by the *Retrieval Manager* to a local *OHS Translator*. The *OHS* provides the *Retrieval Manager* with a quote in the *extendedServicesResponse* which is returned to the client.

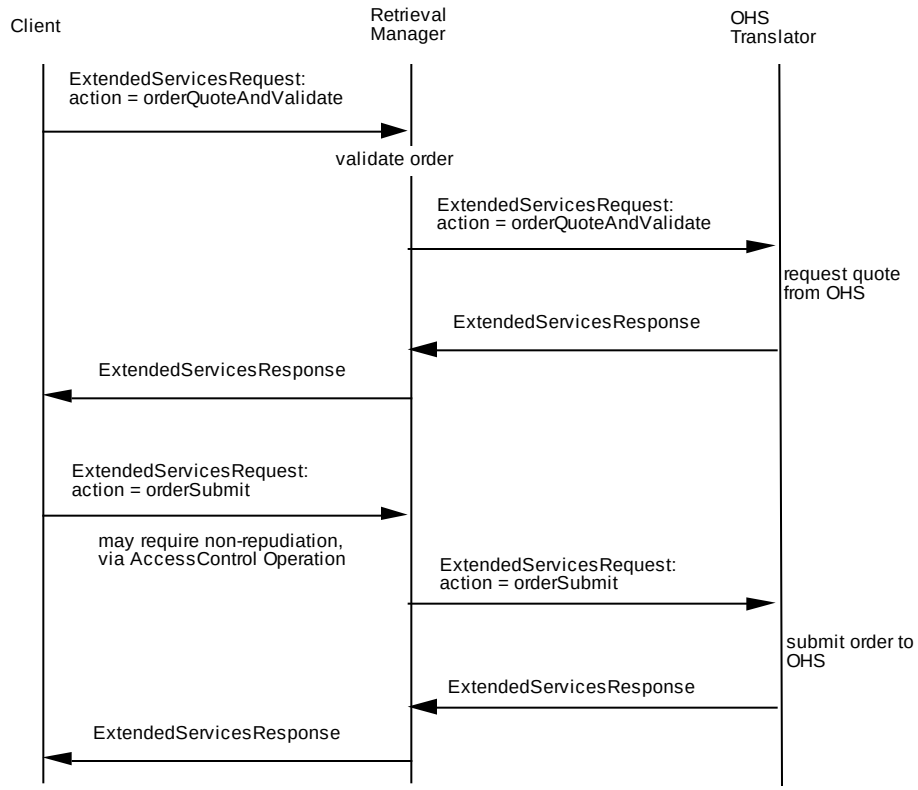

Figure 3-19. Direct Ordering

The scenario shown in Figure 3-19 continues with an order submittal. A user sends a CIP Ordering *extendedServicesRequest,* with the action of *orderSubmit,* when they are committing to the order. In some cases a *Retrieval Manager* will request that the order be submitted in a message which cannot later be denied, i.e. non-repudiation is required on the *orderSubmit*. Non - repudiation requires an *accessControlRequest* message from the *Retrieval Manager* to the client and an *accessControlResponse* message back from the client (not shown in Figure 3-19). The *Retrieval Manager* then submits the order using the *extendedServicesRequest* to the *OHS Translator*. An *extendedServicesResponse* is returned to the *Retrieval Manager* which in turn sends an *extendedServicesResponse* to the client.

The client may specify that the *extendedServicesRequest* operation be executed asynchronously with the *dontWait* option. This option is useful for cases where the client may not wish to wait for the *Task Package* to complete execution. In this case, the *extendedServicesResponse*

indicates either acceptance of the requested *task* or rejection of the *task*. Execution of the *extendedServicesRequest* may take place outside the current CIP Session and the client then needs to query the *Retrieval Manager* with an Order Monitor request to retrieve status information.

Order status in ICS is a pull process, i.e., the current status of an order is pulled from the OHS whenever the user performs an Order Monitor request. Before this request, the order status contained in the Task Package in the *Retrieval Manager's* Extended Services Database may not necessarily be the most current status and hence Search and Present requests against this database are not supported. The *Retrieval Manager* updates the Task Package only when the client explicitly requests an update with an Order Monitor extended service request.

3.3.3.2.2 Indirect Ordering

ICS Compatability: MAA

An indirect ordering scenario by a user involves a *primary order*, a *secondary order* between a local *Retrieval Manager* and a remote *Retrieval Manager*, and another secondary order between the remote *Retrieval Manager* and the OHS *Translator*.

An indirect ordering scenario may have the local *Retrieval Manager* acting as either a proxy or a passthrough. In the proxy case, access rights of the Local *Retrieval Manager* on the Remote *Retrieval Manager* are used to order from the Remote *Retrieval Manager*. This allows a user to order *products* when the user has no privileges with the Remote *Retrieval Manager*. In the pass-through case, the Local *Retrieval Manager* routes the information to the remote *Retrieval Manager*, but the order is requested using the privileges of the user.

The messages passed for an Indirect Order (Figure 3-20) are very similar to the direct ordering case. The major difference is that when the client submits an *extendedServicesResponse* with action of either *orderQuoteAndValidate* or *orderSubmit* to the *Retrieval Manager*, the *Retrieval Manager* identifies that the order is to be filled by an *OHS* to which it does not have direct access. The *Retrieval Manager* then becomes an intermediary and passes the requests off to a Remote *Retrieval Manager*. Both the local and the remote *Retrieval Manager* will create a task package associated with the order request that it receives from the client or local *Retrieval Manager*, respectively.

Asynchronous operations take place as described in the Direct Ordering case.

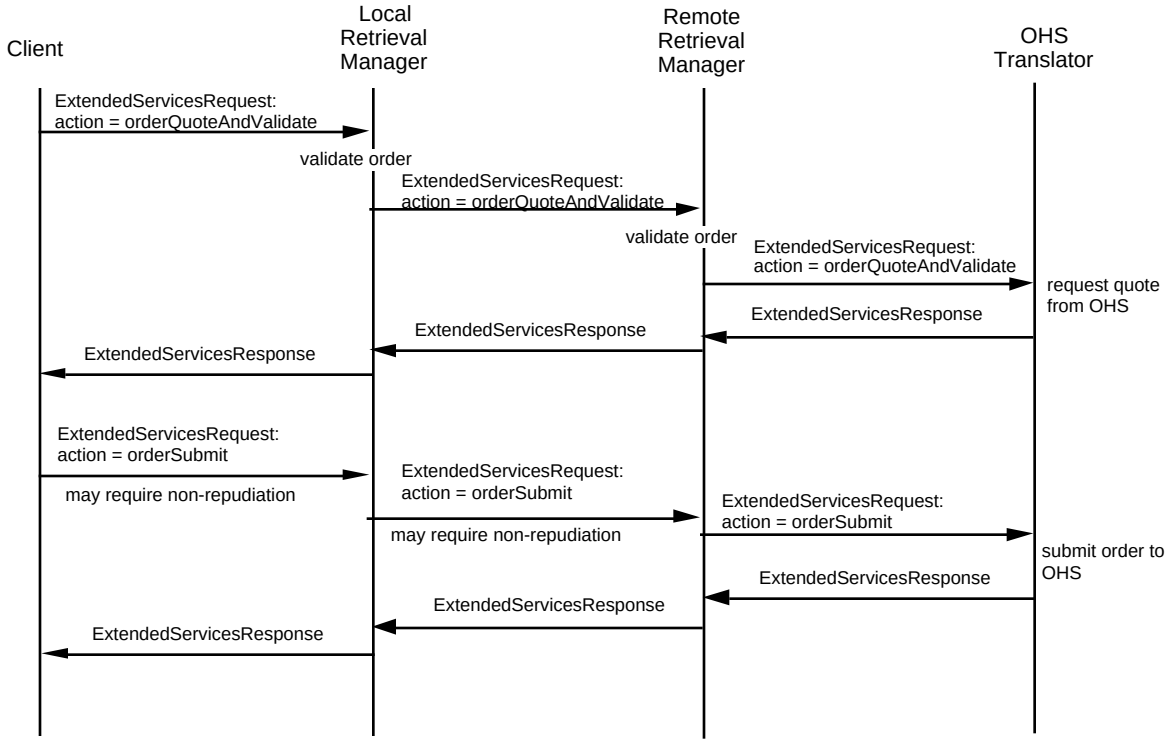

Figure 3-20. Indirect Ordering

3.4 ICS Guide Protocol (IGP) Domain Design

This section provides an overview of the design approach for incorporating the ICS Guide Protocol into ICS when a site needs to manage and access documents. The protocol integrates CIP and HTML document spaces, thereby allowing direct access to documents via distributed HTML gateways and minimizingthe direct impact on document authors. It should be noted that the while the ICS framework supports the integration of this protocol, it only needs to be implemented when the site has documents, such as guide data, that need to be associated and viewed as part of a collection. The ICS Guide Design and Protocol Specification [R24] describes the terminology, goals, and detailed designfor inserting documents into the system, managingthe related metadata, searchingfor documents, and retrievingthe documents.

Document services include both search and present services of the metadata and document content. A document service can be characterized by the following:

- document data can be searched through pre-defined metadata using Attribute/Keyword pairs;
- document data can also be searched by free text search es;
- documents can be retrieved through a known URL ; and
- document data can be presented as metadata or as a textual document .

3.4.1 IGP System Design

The ICS Guide Protocol (IGP) incorporates use of a virtual document that can be used both inside and outside of the ICS document system. This approach has been selected because it permits both ICS internal indexing to provide for fielded and free-text searching within ICS and it allows general-purpose Web search engines to index and access ICS virtual documents. For a detailed discussion on the methodology for using general-purpose Web search engines to access ICS virtual documents, please refer to the ICS Guide Design and Protocol Specification [R24]. The following paragraphs describe Figure 3-21 which shows the ICS site architecture that permits insertion and indexing of documents as well as searching for and retrieval of the virtual documents by the ICS Client.

A "virtual document" is an HTML document generated by software for ICS. It is not stored anywhere, but is only served up at the time the software runs. The virtual document consists of the complete, original, document with necessary information added to it by the software, making it indexable and linking it to CIP collection information. Virtual documents are accessed via a CGI script URL which will point to the software that creates them on-the-fly. Indexing information will be embedded invisibly using HTML META tags.

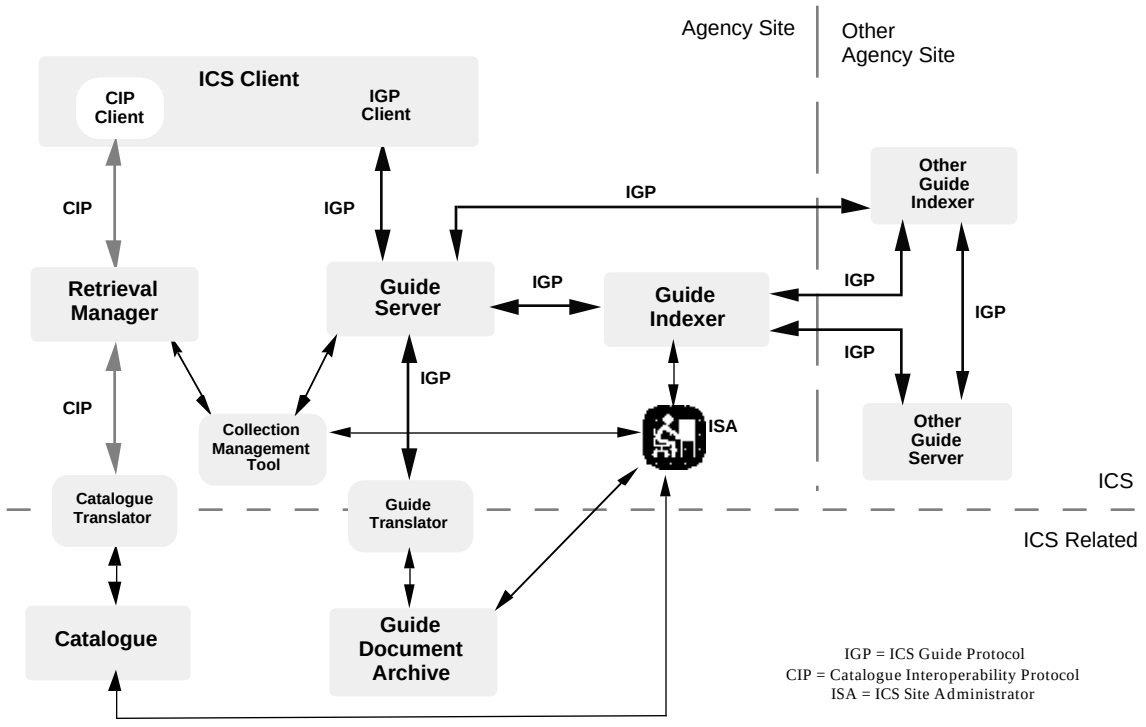

Figure 3-21. ICS Document Management Framework

3.4.1.1 Guide Insertion and Indexing

ICS Compatability: Explanatory

Two custom indexes, a general free-text index and a fielded index, are managed by the ICS Guide Indexer to permit ICS Clients to perform free-text searches and fielded document searches using CIP Attribute/Keyword pairs against ICS-only documents. These indices reference all documents known to each agency site that is part of ICS. Documents can be added to the system either in a "batch" mode, where a whole set of documents is used to establish a new set of indices, or in an "add" mode, where the existing indices are updated with one or more new documents. Batch mode is usually used when a new site is being added to ICS. The batch mode is executed to create the indices for all of the other (existing) agency sites. Add mode is then used for the documents that "belong" to the new site. Add mode is also used when new documents are added to an existing site.

When documents are indexed in batch mode, the ICS Guide Indexer (GI), a specialized Web-crawling index engine, creates the general free-text index and the fielded index for all physical documents known to each agency represented in the ICS. For example, suppose we had an ICS that had two site agencies, the "PDQ" agency and the "XYZ" agency. A new site, agency "ABC" is to be added. First, the batch mode is run to create the indices for agencies PDQ and XYZ at the ABC agency site. To perform this task, the GI starts the indexing by pointing at a top-level Web page that contains hyperlinks to all known agency site's *icsdoc* scripts. This Web page is mirrored at each ICS site. In order to run the *icsdoc* script, the GI requests that the Guide Server (GS) invoke the Guide Translator. The Guide Translator (GT) runs the *icsdoc* script to perform two separate tasks that are discussed in the following paragraphs.

When each *icsdoc* script is invoked, it returns a "directory" that is an HTML document consisting of hyperlinks to all documents reachable through the agency site, one link per line. The hyperlink to each document is NOT the URL of the physical document, but instead is a link back to the *icsdoc* script with path information to the document. Because the Guide Document Archive is empty for the ABC agency, there are no paths returned back from the GT through the GS to the GI. When the GI looks at the next *icsdoc* script URL on the Web page (icsdoc.html), it issues a batch mode request to GS at the XYZ agency site to create the portion of the directory that represents all of the paths to the physical files at that site. As discussed above, the GS at the XYZ agency site requests that its GT create the directory from the Guide Document Archive at the XYZ agency.

Then the ICS Guide Indexer follows these links. Via the Guide Server, the GI requests the Guide Translator to run the *icsdoc* script for each document, with the document path information sent into the script as a parameter; this information identifies a particular physical document that is located in the Guide Document Archive. In this case, the script returns a "virtual document", generated from the physical document referred to in the path information. For example,

contacting the URL http://xyz.org/icsdoc/docs/erbe.html causes *icsdoc* to read and modify the erbe.html document and then return it as a virtual document. This virtual document contains all the HTML of the physical document, plus invisible CIP Attribute/Keyword pairs and a visible link to the appropriate data collection(s) that is created using information from a Guide to Collections mapping file that is accessible to the GT.

Next, the GI scans the virtual document for words and adds the document link to its database, with the words as keys. The virtual document is not stored at the index, just the link. All of the documents in the xyz and pdq "directory" are indexed in the same way to complete indexing for batch mode.

In addition, the GI manages updating of document metadata. When the GT creates the virtual document, it creates as much of the document metadata as it can from the document and performs the mapping between the local schema attributes and CIP attributes. This metadata is returned back to the GI, who then requests the DS to "fill in" any missing attribute information. The GS maintains a global attributes default file that provides information about mandatory CIP attributes and their default values. For cases where the metadata that is created from the virtual document is missing mandatory attributes, the GS supplies the missing attribute names and their default values. As a final "third" pass, the GI requests the GS to display a screen via the ICS Client containing the metadata derived from the document along with the default metadata that has been supplied by the GS. This gives the Site Administrator the opportunity to change any metadata that is deemed appropriate. Lastly, the GI requests that the GS store the document metadata.

After the free-text and fielded indices are created for the XYZ and PDQ agencies, the documents belonging to the new agency site ABC can be added. First, the ABC site manager adds the new documents to the Guide Document Archive shown in Figure 3-21. Or, the document can be made visible to the Guide Document Archive if it is not located at the site. Then, the ABC site manager uses a browser to tell the ICS Guide Indexer the URL for the new document. For example, the URL may be http://abc.org/icsdoc/newdoc.html. The GI then runs its Web-crawling index engine on the URL. Through the Guide Server, the GI requests that the Guide Translator retrieve the document from the Guide Document Archive and run the *icsdoc* script on it, because "icsdoc" is part of the URL. Just as in the batch mode example, the *icsdoc* script creates a virtual document and updates the fielded and free-text with searchable keywords from the document and the virtual document URL. In addition, the document metadata is updated as discussed previously.

Next, as shown in Figure 3-21, the ICS Guide Indexer notifies Guide Indexers at the other ICS agency sites (XYZ and PDQ in our example) that are represented in the icsdoc.html file (i.e., http://abc.org/icsdoc.html) to run a "add" mode request for agency site ABC with the URL that

was specified to the browser. When the other Guide Indexers receive the add request, they proceed in exactly the manner specified for the ABC agency site, but do not notify other Guide Indexers. In this manner, all agency sites maintain up-to-date indices.

3.4.1.2 Guide Searching

ICS Compatability: Explanatory

Free-text and fielded searches are performed via requests issued from the ICS HTTP Client to the ICS Guide Server (GS). Search requests issued to the GS return a list of URL links to the ICS virtual documents that match the query criteria specified by the ICS Client.

For example, suppose a user issues a free-text search request through the ICS Client that requests all documents containing the term “OZONE GSFC”. The URL for the ICS free-text search request would look like the following for the ABC agency site:

http://abc.org/icssearch/free_text="OZONE GSFC"

The GS issues a search request against the free-text index to find the documents that have the term “OZONE GSFC” in them. The URLs for any documents that have that term in them are then returned to the GS. In turn, the GS returns the list of the URLs to the ICS Client for display to the user.

When the user desires to perform a fielded search, ICS Client requests a list of the valid Use attributes from the GS. The GS returns the names of the valid CIP attributes from the global attributes defaults file that can be used for searching. The ICS Client generates a form with empty text boxes for each CIP attribute. Suppose the user enters the following values in these fields:

Author: "John Doe"
PublishingDate: "June 1997"

The search URL would look like:

http://abc.org/icssearch?Author=JohnDoe&
PublishingDate=June1997

When the user submits the form from the ICS Client, the Guide Server will check the fielded index for document URLs that have the specified CIP Attributes. Finally, note that the system can provide both an ICS free-text and fielded search simultaneously. A text field on the fielded search form could be used to enter free text.

As in the free-text search, when the user selects one of the document links, they retrieve the virtual document served by the *icsdoc* script at the appropriate agency site. Then, *icssearch* would check for matching document URLs in the fielded index and the free text index.

3.4.1.3 Guide Document Retrieval

ICS Compatability: Explanatory

When the document search request returns a list of matching document URLs to the ICS Client, the user can select to retrieve a given document and its associated metadata. Upon selection, the ICS Client issues a document retrieval request to the Guide Server. The GS retrieves the metadata for the document from the Guide Metadata Database and requests the Guide Translator to create the virtual document using the *icsdoc* script and the URL selected by the user. The GT creates the document and it is made available to the user via the HTTP Client that is part of the ICS Client.

3.4.2 IGP Protocol Overview

ICS Compatability: Explanatory (See R24 for IGP Compliance)

IGP messages are exchanged between the ICS Client, Guide Server, Guide Indexer, and Guide Translator in order to manage document storage and retrieval within ICS. The messages which the IGP uses are listed Table 3-2. The messages are used in the SDD to demonstrated the use of IGP in ICS. The ICS Guide Design and Protocol Specification [R24] provides a controlled definition of the contents and meaning of each of the messages in Table 3-2.

Table 3-2. IGP Messages (1 of 2)

Message Name	How It Is Used
FieldSearchForm	ICS Client sends this message to ICS Guide Server, which runs *icssearch* script. This message causes script to read Attributes Default File, generate a fielded search form, and returns form to client to display.
GetCollection	Link from document to collection for an ICS Client. CIP Client returns collection search results to CIP Client.
GetCollectionGuest	Link from document to collection for a general Web browser. Links to CIP Client. Starts a session, shows collection search results.
GetDirectory	Indexing engines send this message to ICS Guide Server, which has Guide Translator run the *icsdoc* script without path information. Returns a list of URLs of all virtual documents reachable through the site.
GetVirtualDocument	Indexing engines, browsers, and ICS Clients send this message to ICS Guide Server, which has Guide Translator run the *icsdoc* script with the path information to a document. Returns a virtual document. ICS Clients append the "&ICS_CLIENT" to notify the Guide Translator which type of collection link to insert.
IndexAdd	ICS Guide Indexer sends this message to the ICS Guide Indexer at other sites, which instructs them to update their indexes with the new guide document.

Table 3-2. IGP Messages (2 of 2)

Message Name	How It Is Used
IndexNew	An ICS Site Administrator sends this message manually (or browser client software sends it) to ICS Guide Indexer at his site, notifying it of a new guide to be indexed at the site.
IndexValues	ICS Site Administrator browser sends this message to ICS Guide Indexer to update the Guide Metadata Database with the information provided. This is done during guide ingest.
SearchFields	ICS Client sends this message to ICS Guide Server, which runs *icssearch* script. This queries the fielded index with the supplied values for specified attributes. If more than one attribute/value pair is specified, an "&" appears before each additional pair. Returns a list of URLs of virtual documents that have at least one pair in its entry in the Guide Metadata Database. (Note this implies logical OR between pairs). A search of both fielded and free-text indexes can be done by using "free_text" as one of the attributes and the text as the value.
SearchFreeText	ICS Client sends this message to ICS Guide Server, which runs *icssearch* script. This queries the free-text index with the supplied text. Returns a list of URLs of virtual documents that contain the text.
UpdateCollections	Collection Management Tool sends this message to ICS Guide Server when the ISA adds a guide_url to a collection in the Collections Data Base. The server has the Guide Translator run *ics_collection_update* script, which updates Collections Mapping File.
UpdateMetadata	ICS Guide Index sends this message to ICS Guide Server when it has Guide Attribute/Value pairs for a new document added to the system. The server has the Guide Translator run *ics_metadata_update* script, which updates the Guide Metadata Database.

3.4.3 IGP Operations

Messages are passed between ICS Guide elements primarily using IGP. IGP is based on the HTTP and is used to pass primarily indexing, search and guide retrieval requests. This section shows how various messages are passed between the distributed ICS elements to accomplish a request by a user for a search or a guide retrieval.

3.4.3.1 ICS Fielded Guide Search

ICS Compatability: MAA

Figure 3-22 shows the two sets of messages that are used to accomplish an ICS fielded guide search. First, the *ICS Client* issues the *FieldSearchFormRequest* to the *Guide Server* to create a fielded search form from the Attributes Default File. The form is passed back to the *ICS Client* via the *FieldSearchFormResponse* so that it can be displayed to the user. The user then specifies values for the desired search fields and the *ICS Client* issues a *SearchFieldsRequest* to the *Guide Server*. The *Guide Server* searches the fielded index to determine if there are any attribute/value matches. If so, the *Guide Server* returns a list of URLs of virtual documents that contain the

matching attribute/value pairs. Although not depicted in Figure 3-22, it should be noted that both a fielded search and free-text search can be performed via the *SearchFieldsRequest*. In order to perform this, the user specifies "free_text" as one of the attributes and the search text as the value. The *Guide Server* then searches the free-text index in addition to the fielded index.

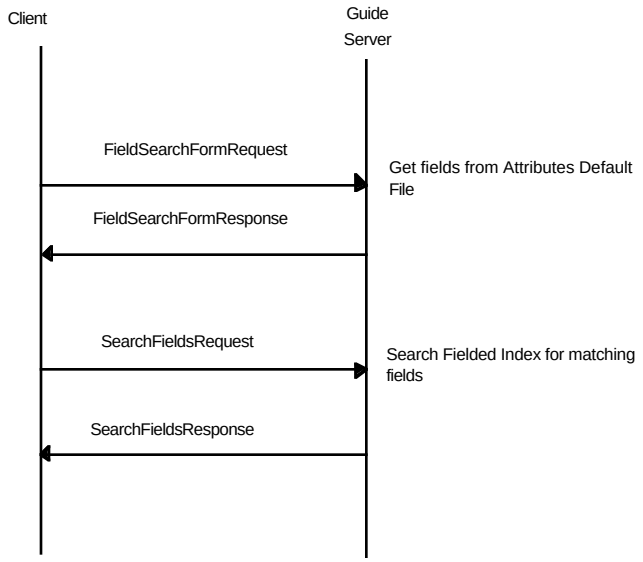

Figure 3-22. Fielded Guide Search

3.4.3.2 ICS Client Free-Text Guide Search

ICS Compatability: MAA

Free-text searches for guide documents are performed in ICS using the messages depicted in Figure 3-23. The *ICS Client* presents a simple form with a single text box to the user. The user enters text that must be in the list of guides that will be returned. When the user submits the form, the *ICS Client* issues a *SearchFreeTextRequest* to the *Guide Server*. The *Guide Server* then searches the free-text index with the supplied text to determine if there is a match. The list of URLs of virtual documents that contain the text is then returned to the *ICS Client* via the *SearchFreeTextReponse* message.

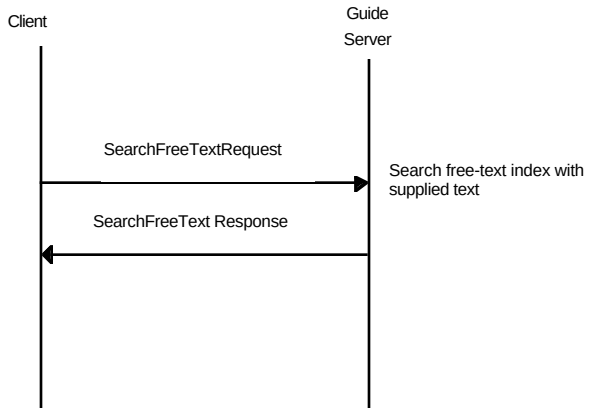

Figure 3-23. ICS Client Free-Text Guide Search

3.4.3.3 ICS Guide Retrieval from ICS Client

ICS Compatability: MAA

Retrieval of virtual guide documents is performed in ICS using the messages shown in Figure 3-24. Based on a list of URLs presented from either the fielded or free-text searches, the user can specify the URL of a virtual guide document for retrieval. The ICS Client passes the URL via the GetVirtualDocumentRequest to the GuideServer. The Guide Server routes the message to the Guide Translator to create the virtual document. The Guide Translator retrieves the Guide Document from the local store and retrieves the metadata from the Guide Metadata Database so that it can construct Meta tags and links for the related collections in the virtual document. It should be noted that each site contains free-text and fielded indices for all sites within ICS. If the user at site A requests retrieval of a document from site B, then it is the *Guide Server* and *Guide Translator* from site B that actually performs the retrieval. Once the virtual document is created by the *Guide Translator*, it is returned to the *ICS Client* via the *Guide Server* in the *GetVirtualDocumentResponse.*

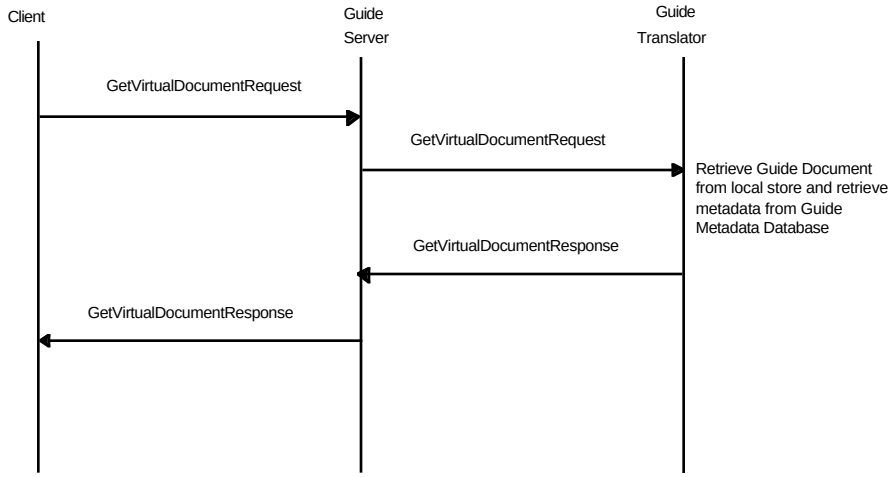

Figure 3-24. ICS Guide Retrieval from ICS Client

3.5 Identification of ICS Element Services and Interfaces

This section provides an identification of the services provided by each element shown in the ICS Framework. The remainder of this section provides the identification of services for the following elements:

- *Retrieval Manager*
- *ICS Client*
- *Catalogue Translator*
- *OHS Translator*
- *UPS Translator*
- *HTTP/CIP Gateway*
- *ICS Gateway*
- *ICS Site Administrator (ISA)*
- *Collection Management Tool (CMT)*
- *Monitoring and Control Tool (MCT)*
- *Guide Server*
- *Guide Indexer*
- *Guide Translator*

Section 3-6 describes services assumed to be provided by the non-ICS elements.

The services supplied by a given site, using ICS elements, will vary by site. The services in this section are defined assuming a maximum site. The minimum services needed to be provided by a site to be ICS Compatible are described in Section 9.

3.5.1 Retrieval Manager Services

ICS Compatability: MAA

The following services are provided by the *Retrieval Manager*:

- CIP Compatible Messaging based on Z39.50 messaging and session management: *origin* and *target*
- Session Management: Manages user requests including; queries and associated sub-queries, search routing, recursive search trapping, result set collation and presenting, order management.
- Collection Management: persistent storage of local *collection*; services to create, ingest and maintain local *collection*; attribute management; promotion of hot *collections* to theme *collections*.
- Explain Database: Persistent Store of Explain data
- Native Extended Services: Creation, modification and persistent storage of *Task Package* in response to *Extended Service* requests and responses.
- CIP Ordering Extended Services: A CIP customized Extended Service for ordering EO Products. Specific CIP Order Extended Service request types are Order Validation and Quotation, Order Submission, Order Monitoring and Order Cancellation.
- User Management: persistent storage of user information where user may be a human using a client or may be another *Retrieval Manager*.
- Operator Interface for *ICS Site Administrator* administration of the *Retrieval Manager*.
- Monitoring and Error Management: User session errors, CIP diagnostic messages, network diagnostics messages
- E-mail Client: E-mail to *ISA* for error messages.

The *Retrieval Manager* design is required to be a modular design allowing a site to disable specific services which are not be hosted at the site. The variation of *Retrieval Manager* services across sites is described in Section 3.3.1.2.

The *Retrieval Manager* has the functional interfaces indicated in Table 3-3.

Table 3-3. Retrieval Manager Interfaces (1 of 2)

Other ICS or Related Element	Interface Description
ICS Client	A *Retrieval Manager* is capable of forming *CIP sessions* with multiple *ICS Clients* where the *ICS Client* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Other z39.50 Clients	*Retrieval Manager* is capable of forming a z39.50 Z-association with other z39.50 clients, where the *Other Z39.50 Client* is the *origin* and *Retrieval Manager* is the *target*. All Z39.50, Version 2 and 3 sessions are supported by this interface with the exception of the following z39.50-1995 services: *scanRequest, scanResponse, sortRequest, sortResponse, extendedServicesRequest, extendedServicesResponse.*
Other *Retrieval Manager*s	A *Retrieval Manager* is capable of forming CIP sessions with other *Retrieval Manager*s where either *Retrieval Manager* may be the *origin* or the *target*. All CIP messages are supported by this interface.
ICS Gateway	A *Retrieval Manager* is capable of forming *CIP sessions* with multiple *ICS Gateways* where the *ICS Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Catalogue Translator	A *Retrieval Manager* is capable of forming CIP sessions with multiple Catalogue *Translator*s where the *Retrieval Manager* is the *origin* and the Catalogue *Translator* is the *target*. All CIP messages are supported by this interface with the exception that CIP Ordering Extended Service is not supported by this interface. (Note that other interface approaches are under investigation, e.g., direct function call, for those sites with tight coupling between the *Retrieval Manager* and *Translator*.)
OHS Translator	A *Retrieval Manager* is capable of forming CIP sessions with multiple OHS *Translator*s where the *Retrieval Manager* is the *origin* and the OHS *Translator* is the *target*. The following CIP messages are supported by this interface: *InitializeRequest, InitializeResponse, AccessControlRequest, AccessControlResponse, ExtendedServicesRequest, ExtendedServicesResponse, Close.* The only valid extended services type for this interface is CIP ordering

Table 3-3. Retrieval Manager Interfaces (2 of 2)

Other ICS or Related Element	Interface Description
UPS Translator	A *Retrieval Manager* may have an interface to a UPS *Translator* to request User Management Data including User Authentication Information. This interface may be accomplished using CIP. A Retrieval Manager may have an interface to a Certification Authority to perform public key directory lookups. This interface should use the X.509 Lightweight Directory Access Protocol (LDAP) [R23].
ICS Site Administrator (ISA)	The *Retrieval Manager* provides an operator interface to the *ISA*. The operator interface provides a graphical user interface to allow the *ISA* to monitor, control, diagnose, and maintain the operations of the *Retrieval Manager*.
Collection Management Tool	The *Retrieval Manager* allows the *CMT* to modify the EO content of the *Retrieval Manager Collection Data Base*
Monitoring and Control Tool	The *Retrieval Manager* allows an *MCT* interface to gather operation and error handling data from the *Retrieval Manager* and to responds to *MCT* commands for controlling the configuration and state of *Retrieval Manager* processes. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.5.2 ICS Client Services

ICS Compatability: MAA

The following services are provided by an *ICS Client*:

- CIP Compatible Messaging. The *ICS Client* is able to send and receive all CIP messages.
- Search Formulation. The *ICS Client* supports the user in formulating queries, e.g., queries of *collection*s and *products*, incremental queries, conversion of geographic inputs into CIP queries.
- Result Set handling. The *ICS Client* supports the user in viewing temporary result sets, and conversion of a temporary result set to a persistent result set.
- Order Formulation. The *ICS Client* supports the user in specifying and reviewing quotes on orders, including conversion of a multi-site result set into single site orders.
- Authentication Support. The *ICS Client* supports the user in providing the user's credentials to form an authenticated session or to respond to an authentication request from the *Retrieval Manager*.
- Dynamic Configuration. The *ICS Client* supports the retrieval and caching of explain information from *Retrieval Managers* including definitions for attributes and record structures. The *ICS Client* provides the attributes in support of a user creating searches.
- User Profile Management. The *ICS Client* may support the persistent storage of user profile data.
- Document Search Formulation. The *ICS Client* supports the user in formulating free-text and fielded searches for documents within ICS.

Note that the list above intentionally does not include presentation services to the user.

The *ICS Client* has the functional interfaces indicated in Table 3-4.

Table 3-4. ICS Client Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	*ICS Client* is capable of forming CIP sessions with at least one *Retrieval Manager* where the ICS Client is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
ICS Guide Server	*ICS Client* is capable of forming an HTTP connection using IGP messages with the *ICS Guide Server* where the ICS Client is the client and the *ICS Guide Server* is the server. All IGP messages are supported by this interface.
Interfaces outside of ICS	The *ICS Client* may have interfaces to the following ICS related elements: the *Archive*. The *ICS Client* may access the *Archive* using ftp to transfer files from the data store for user access through the ICS Client.

3.5.3 Catalogue Translator Services

ICS Compatability: MAA

The following services are provided by the *Catalogue Translator*. Other CIP Translator configurations are described in Section 3.2.2.3. For example, if a single CIP Translator is used to interface to a data provider (see the "Retrieval Manager as a Catalogue Gateway" example in Figure 3-12) the services of a *Catalogue Translator, OHS Translator* and *UPS Translator* are merged.

- mapping of CIP attributes to the attributes of the local Catalogue (i.e., for product and guide)
- mapping of CIP query elements to local query elements
- mapping of local result sets into CIP result sets
- use of the local catalogue communication protocol
- query optimization

The *Translator* contains elements which are common across ICS sites, e.g., those parts of the *Translator* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with the *Catalogue.*

The *Catalogue Translator* has the functional interfaces indicated in Table 3-5.

Table 3-5. Catalogue Translator Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	A Catalogue *Translator* is capable of being a CIP *target* for *CIP sessions* from a single *origin*. Typically the *origin* will be a *Retrieval Manager*. All CIP messages are supported by this interface except the CIP Ordering Extended Service. A catalogue translator will typically not support *collection* seaches or Explain searches. (Note that other interface approaches are under investigation, e.g., direct function call, for those sites with tight coupling between the *Retrieval Manager* and *translator*.)
Interfaces outside of ICS	The *Catalogue Translator* interfaces to the following ICS related elements: *Catalogue*

3.5.4 OHS Translator Services

ICS Compatability: MAA

The following services are provided by the *OHS Translator*.

- Convert CIP order objects to local order handling system objects
- use of the local *catalogue* communication protocol

The *Translator* contains elements which are common across ICS sites, e.g., those parts of the *Translator* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with the *OHS*.

The *OHS Translator* has the functional interfaces indicated in Table 3-6.

Table 3-6. OHS Translator Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	An *OHS Translator* is capable of being a CIP *target* for one CIP session. Typically the *origin* will be a Retrieval Manager. The following CIP messages are supported by this interface: *InitializeRequest, InitializeResponse, AccessControlRequest, AccessControlResponse, ExtendedServicesRequest, ExtendedServicesResponse, Close*. The only valid extended services type for this interface is CIP ordering
Interfaces outside of ICS	The *OHS Translator* interfaces to the following ICS related elements: *Order Handling System*

3.5.5 UPS Translator Services

ICS Compatability: MAA

The following services are provided by the *UPS Translator*.

- Local user profile to CIP user profile
- Secure handling of authentication information
- Use of the local *catalogue* communication protocol for user information

The *Translator* contains elements which are common across ICS sites, e.g., those parts which interface to the *Retrieval Manager*, and contains parts which are unique to the site, e.g., those parts which speak to the local *UPS*.

Multiple *Translator*s may be provided to allow multiple *Retrieval Managers* to interface with a *UPS*.

The *UPS Translator* has the functional interfaces indicated in Table 3-7.

Table 3-7. UPS Translator Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	An *UPS Translator* is capable of being a CIP *target* for one *CIP session*. Typically the *origin* will be a Retrieval Manager. This interface may be accomplished using CIP.
Interfaces outside of ICS	The *UPS Translator* interfaces to the following ICS related element: *User Profile System*

3.5.6 HTTP/CIP Gateway Services

ICS Compatability: MAA

The following services are provided by the *HTTP/CIP Gateway*:

- CIP Messaging. The *HTTP/CIP Gateway* converts the user's inputs received via HTTP to CIP messages for transmission to a *Retrieval Manager*.
- HTML Generation. The *HTTP/CIP Gateway* dynamically generates HTML output from a set of templates for presentation of CIP information to the user. This may be based on CGI scripts (programs) capable of receiving input from - and sending HTML output to - any standard WWW browser.
- Session management. The *HTTP/CIP Gateway* maintains CIP session context in response to http messages which is a state-less protocol, e.g., preservation of information from screen to screen for a particular user session (e.g., result set IDs, user preferences). The HTTP protocol does not maintain session information, it is a one time passing of information. Virtually every single mouse click to a browser results in the gateway generating a single page from scratch.
- Security. The *HTTP/CIP Gateway* supports user authentication in the ICS.

- Explain Data Management. The gateway provides an Explain cache as a local cache for the Explain information provided by the *Retrieval Manager*
- *ICS Client* server. Some *HTTP/CIP Gateway*s may act as servers of portable client code. One example is a Java applet which would allow a Java enabled web-browser to run some or most of the *ICS Client* layers at the user's computer.

The *HTTP/CIP Gateway* has the functional interfaces indicated in Table 3-8.

Table 3-8. HTTP/CIP Gateway Application Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	*HTTP/CIP Gateway* is capable of forming CIP sessions with at least one *Retrieval Manager* where the *HTTP/CIP Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Interfaces outside of ICS	The *HTTP/CIP Gateway* may have interfaces to the following ICS related elements: *a web server*

3.5.7 ICS Gateway

ICS Compatability: MAA

The following services are provided by the *ICS Gateway*.

- mapping of local attributes to CIP attributes (e.g. for *product*, *guide* and *collection*)
- mapping of local query elements to CIP query elements
- mapping of CIP result sets into local result sets
- use of the local *catalogue* communication protocol
- The *ICS Gateway* converts the user's inputs received via local protocol to CIP messages for the *Retrieval Manager*.

The *ICS Gateway* contains elements which are common across ICS sites, e.g., those parts of the *ICS Gateway* which speak CIP, and contains parts which are unique to the site, e.g., those parts which speak the local protocol.

The *ICS Gateway* has the functional interfaces indicated in Table 3-9.

Table 3-9. ICS Gateway Application Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	*ICS Gateway* is capable of forming *CIP sessions* with at least one *Retrieval Manager* where the *ICS Gateway* is the *origin* and the *Retrieval Manager* is the *target*. All CIP messages are supported by this interface.
Interfaces outside of ICS	The *ICS Gateway* may have interfaces to the following ICS related elements: *catalogue* and *Existing agency client.*

3.5.8 ICS Site Administrator (ISA) Operations

ICS Compatability: MAA

The following operations are performed by the *ISA*:

- Collection Definition. The *ISA* creates the site *collections* in accordance with the ICS Collection Manual, including site *provider archive collections*, *provider theme collections*, links to *collections* held at other sites.
- Collection Maintenance. The *ISA* maintains the site *collections* in accordance with the ICS Collection Manual, including periodically checking the *collections* held in their *Retrieval Manager*s for consistency.
- *Retrieval Manager* data. The ISA maintains all data in the *Retrieval Manager* databases.
- *Retrieval Manager* Operations. The *ISA* monitors and corrects any incorrect operations of the *Retrieval Manager* at their site.
- Guide Document Definition. The *ISA* adds Guide Documents to ICS and updates related metadata.

The *ISA* has the operational interfaces indicated in Table 3-10.

Table 3-10. ISA Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	The *ISA* accesses the *Retrieval Manager* via an operator's interface. The operator interface provides a graphical user interface to allow the *ISA* to monitor, control, diagnose, and maintain the operations of the *Retrieval Manager*.
Collection Management Tool	The *ISA* uses a *CMT* operator interface to control ingest of data into the *CMT*, modification of the data, and insertion of CIP *Collections* into the *Collection Database*.
Catalogue Translator	The *ISA* maintains the data and monitors the operations of the *Catalogue Translator* at the site.
OHS Translator	The *ISA* maintains the data and monitors the operations of the *OHS Translator* at the site.
UPS Translator	The *ISA* maintains the data and monitors the operations of the *UPS Translator* at the site.
Guide Indexer	The *ISA* accesses the *Guide Indexer* via an operator's interface. The operator interface provides a graphical user interface to allow the *ISA* to add guide documents, update related metadata, or request batch update of the document indices.
Interfaces outside of ICS	The *ISA* may have interfaces to the following ICS related elements: *OHS, UPS, Catalogue, Archive, Guide Document Archive*. These interfaces are not discussed here as they are outside of ICS.

3.5.9 Collection Management Tool (CMT) Services

ICS Compatability: MAA

The following services are provided by the *CMT*:

- Conversion of files into CIP compatible format *Collections*
- Support maintenance of the *Collection Data Base*
- Provides mechanism to define linkage between *Collections* and *Guide Documents*

The *CMT* has the functional interfaces indicated in Table 3-11.

Table 3-11. Collection Management Tool Interfaces

Other ICS or Related Element	Interface Description
*Retrieval Manager*s	The *CMT* interfaces to the *Collection Database and Explain Database* of the *Retrieval Manager* to establish and maintain the EO content of the *collections* held in the data bases.
ISA	The *CMT* provides an operator interface to the *ISA*. The operator interface allows the *ISA* to control ingest of data into the *CMT*, modification of the data, and insertion of CIP *Collections* into the *Retrieval Manager*.
Guide Server	The *CMT* notifies the Guide Server with an *UpdateCollections* message so that the Collections Mapping File can be updated.
Interfaces outside of ICS	The *CMT* will have an interface to accept data which is to be made into CIP *collections*, e.g. EOSDIS V0 DAAC data set descriptions, DIFs. This interfaces is not discussed here as it is outside of ICS.

3.5.10 Monitoring and Control Tools (MCT) Services

ICS Compatability: MAA

The following services are provided by the *MCT*:

- Monitoring of *Retrieval Manager* operations and error handling.
- Commanding the *Retrieval Manager* to change states based on SSM commands

The *MCT* has the functional interfaces indicated in Table 3-12.

Table 3-12. Monitoring and Control Tool Interfaces

Other ICS or Related Element	Interface Description
Retrieval Manager	The *MCT* interfaces to the Retrieval Manager to gather operation and error handling data from the Retrieval Manager and to control the configuration and state of Retrieval Manager processes. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.
Translators	The *MCT* may interface to the translators to gather status and command changes similar to the interface with the *Retrieval Manager*. This particular interface is TBS.
Interfaces outside of ICS	The *MCT* may have interfaces to the following ICS related element: SSM. These interfaces are not discussed here as they are outside of ICS. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.5.11 Guide Server Services

ICS Compatability: MAA

The following services are provided by the *Guide Server*:

- Manages free-text and fielded searches for guide documents
- Includes an HTTP Daemon
- Manages retrieval service for virtual guide documents
- Holds the Guide Metadata attribute defaults
- Holds the Guide Metadata Database

The *ICS Guide Server* has the functional interfaces indicated in Table 3-13.

Table 3-13. ICS Guide Server Interfaces

Other ICS or Related Element	Interface Description
ICS Client	The *Guide Server* is capable of forming IGP connection with the *ICS Client* where the *Guide Server* is the *origin* and the *Guide Server* is the target. This interface is an IGP interface which is based on HTTP
Guide Translator	The *Guide Server* interacts with the *Guide Translator* to pass messages to request Directory searching and retrieval of virtual documents. This interface is an IGP interface which is based on HTTP
Guide Indexer	The *Guide Server* interfaces with the local *Guide Indexer* and *Other Guide Indexers* to update Guide Metadata databases. This interface is an IGP interface which is based on HTTP
CMT	The Guide Server receives an *UpdateCollections* message from the *CMT* so that the Collections Mapping File can be updated.
Interfaces outside of ICS	The *ICS Guide Server* may have interfaces to the following ICS related element: SSM. These interfaces are not discussed here as they are outside of ICS. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.5.12 Guide Indexer Services

ICS Compatability: MAA

The following services are provided by the *ICS Guide Indexer*:

- Creates and updates the ICS Free-Text index of ICS Guide Documents
- Creates and updates the Fielded index of ICS Guide Documents
- Lists all sites of ICS Guide Servers
- Contains an HTTP Daemon

The *Guide Indexer* has the functional interfaces indicated in Table 3-14.

Table 3-14. Guide Indexer Interfaces

Other ICS or Related Element	Interface Description
Guide Server	The *Guide Indexer* interfaces with the *Guide Server* to update the Guide Metadata database.
Other *ICS Guide Indexers*	The *Guide Indexer* interfaces with other *Guide Indexers in ICS* to let them know that a new URL has been added and that their indices need to be updated.
Other *Guide Servers*	The *Guide Indexer* interfaces with other *Guide Servers* to update the Guide Metadata databases.
Interfaces outside of ICS	The *Guide Indexer* may have interfaces to the following ICS related element: SSM. These interfaces are not discussed here as they are outside of ICS. This interface should be based on a standard protocol, e.g., SNMP, to be defined in later releases.

3.5.13 ICS Guide Translator Services

ICS Compatability: MAA

The following services are provided by the *ICS Guide Translator*:

- Gets directory of all URLs to all guide documents that are reachable through the agency site
- Retrieves virtual documents
- Updates collection and Guide Document metadata

The *Guide Translator* has the functional interfaces indicated in Table 3-15.

Table 3-15. ICS Guide Translator Interfaces

Other ICS or Related Element	Interface Description
ICS Guide Server	The *ICS Guide Translator* returns directories for guide documents and virtual documents to the *Guide Server.*
Interfaces outside of ICS	The *Guide Translator* has interfaces to the following ICS related element: *Guide Document Archive.* These interfaces are not discussed here as they are outside of ICS.

3.6 Identification of ICS Related Element Services

This section describes the services assumed to be provided by Elements related to ICS but not part of ICS. For these elements which are shown on the ICS Framework (Figure 3-9) but are not part of the ICS per se, the following identification of services are assumptions which will be used as the ICS development proceeds.

- *Catalogue*
- *Order Handling System (OHS)*
- *User Profile System (UPS)*
- *Archive*
- *Site System Management (SSM)*
- *Guide Document Archive*

3.6.1 Existing Catalogue Services

ICS Compatability: Explanatory

The following services are assumed to be provided by the *Catalogue*:

- Directory service provides descriptions of data sets containing high level information suitable for making an initial determination of the potential usefulness of a data set for some applications. This information will be used to populate ICS *collections.*
- Inventory service provides the information about the data held in an archive. This information will be used as ICS *product* descriptor data.
- Browse service provides access to *browse* data.

Note that a particular existing catalogue may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Catalogue*.

3.6.2 Order Handling System (OHS) Services

ICS Compatability: Explanatory

The following services are assumed to be provided by an existing *OHS*:

- Persistent Storage of Orders originating from a *Retrieval Manager*
- Formulation of quotes on orders
- Statusing of orders
- Billing and accounting of users orders (for orders filled by the *OHS* and by orders for which the local *Retrieval Manager* acted as a proxy.)
- Notification to users of order filled
- Invoices to users

Note that a particular *OHS* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *OHS*.

3.6.3 User Profile System (UPS) Services

ICS Compatability: Explanatory

The following services are assumed to be provided by an existing *UPS*:

- Persistent Storage of user profile data

Note that a particular *UPS* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *UPS*.

For a *UPS* that is based on a public-key infrastructure (PKI), the following services are assumed to be provided by a Certification Authority: (CA)

- Publish the criteria for granting user certificates
- Granting, revoking and general certificate management functions
- storing root keys

3.6.4 Archive Services

ICS Compatability: Explanatory

The following services are assumed to be provided by an *Archive*:

- Persistent Storage of EO data
- Filling of orders: delivery via ftp or media

Note that a particular *Archive* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Archive*.

3.6.5 Site System Management (SSM)

ICS Compatability: Explanatory

The following services are assumed to be provided by an existing *SSM*:

- Management oversight of functional elements at the data provider site
- Monitoring of site services
- Commanding of site service configuration and modes

Note that a particular *Site System Management* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Site System Management*.

3.6.6 Guide Document Archive

ICS Compatability: Explanatory

The following services are assumed to be provided by an existing *Guide Document Archive:*

- Retrieval of documents from within the collection

Note that a particular *Guide Document Archive* may not provide all of these services. But, it is the assumption of the ICS SDD that if these services are provided by a data provider that they are provided by the *Guide Document Archive*.

This page intentionally left blank.